National Library of Canada
Bibliothèque nationale du Canada

Canadian Theses Service
Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

NOTICE

The quality of this microform is heavily dependent upon the quality of the original thesis submitted for microfilming. Every effort has been made to ensure the highest quality of reproduction possible.

If pages are missing, contact the university which granted the degree.

Some pages may have indistinct print especially if the original pages were typed with a poor typewriter ribbon or if the university sent us an inferior photocopy.

Reproduction in full or in part of this microform is governed by the Canadian Copyright Act, R.S.C. 1970, c. C-30, and subsequent amendments.

AVIS

La qualité de cette microforme dépend grandement de la qualité de la thèse soumise au microfilmage. Nous avons tout fait pour assurer une qualité supérieure de reproduction.

S'il manque des pages, veuillez communiquer avec l'université qui a conféré le grade.

La qualité d'impression de certaines pages peut laisser à désirer, surtout si les pages originales ont été dactylographiées à l'aide d'un ruban usé ou si l'université nous a fait parvenir une photocopie de qualité inférieure.

La reproduction, même partielle, de cette microforme est soumise à la Loi canadienne sur le droit d'auteur, SRC 1970, c. C-30, et ses amendements subséquents.

NL-339 (r. 86/04) c

Canada

UNIVERSITY OF ALBERTA

AN OPTOELECTRONIC TRANSVERSAL FILTER

BY

BRENT SWEKLA

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE.

DEPARTMENT OF ELECTRICAL ENGINEERING

EDMONTON, ALBERTA

FALL, 1991

National Library of Canada
Bibliothèque nationale du Canada

Canadian Theses Service
Service des thèses canadiennes

Ottawa, Canada
K1A 0N4

The author has granted an irrevocable non-exclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of his/her thesis by any means and in any form or format, making this thesis available to interested persons.

L'auteur a accordé une licence irrévocable et non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de sa thèse de quelque manière et sous quelque forme que ce soit pour mettre des exemplaires de cette thèse à la disposition des personnes intéressées.

The author retains ownership of the copyright in his/her thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without his/her permission.

L'auteur conserve la propriété du droit d'auteur qui protège sa thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

ISBN 0-315-70043-2

Canada

UNIVERSITY OF ALBERTA

RELEASE FORM

NAME OF AUTHOR: BRENT SWEKLA

TITLE OF THESIS: AN OPTOELECTRONIC TRANSVERSAL FILTER

DEGREE: MASTER OF SCIENCE

YEAR THIS DEGREE GRANTED: 1991

Permission is hereby granted to the University of Alberta Library to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only.

The author reserves all other publication and other rights in association with the copyright in the thesis, and except as hereinbefore provided neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatever without the author's prior written permission.

Brent Swekla

15112 - 49 Ave
Edmonton, Alberta
Canada
T6H 5M8

October 11, 1991

UNIVERSITY OF ALBERTA

FACULTY OF GRADUATE STUDIES AND RESEARCH

The undersigned certify that they have read, and recommend to the Faculty of Graduate Studies and Research for acceptance, a thesis entitled AN OPTOELECTRONIC TRANSVERSAL FILTER submitted by BRENT SWEKLA in partial fulfillment of the requirements for the degree of MASTER OF SCIENCE.

Dr. R. I. MacDonald

Dr. J. N. McMullin

Dr. J. Conradi

Dr. W. Allegretto

Dr. R. P. W. Lawson

October 8, 1991

Abstract

This thesis discusses a novel transversal filter implementation which employs optical fiber delays and optoelectronically set weights.

The wideband, low loss signal propagation which is characteristic of optical fiber makes it an excellent delay medium. Intensity modulated signals can be delayed and combined to produce filters for signals with bandwidths characteristic of optical communications technology, that is, several gigahertz. The principal drawbacks of the optical fiber transversal filters which have been demonstrated to date are their inflexible or expensive schemes for setting the transversal filter tap weights. This thesis describes a novel approach to setting the weights. By adjusting the bias voltage across a gallium arsenide metal-semiconductor-metal photodetector one can control its sensitivity. Bipolar sensitivity can even be achieved by reversing the polarity of the bias, which introduces a 180 degree phase shift in the detector's output.

The present work employs gallium arsenide metal-semiconductor-metal photodetectors to set tap weights in a transversal filter using optical delays. This has advantages in simplicity, size, and cost over other weighting schemes. The thesis describes the technological problems involved in constructing an optoelectronic transversal filter, and how they are solved. The thesis also presents experimental results demonstrating the operation and agility of the completed filter and describes the design, construction, and operation of a multichannel analog voltage controller which was built to facilitate the demonstration of the transversal filter and to be used for future work.

Acknowledgements

I wish to thank Dr. R. I. MacDonald, whose idea that variable sensitivity photodetectors could be used to implement a transversal filter is the basis of this thesis, for his capable supervision and encourgagement. I would also like to thank the members of the examining committee, Drs. J. Conradi, J. N. McMullin, and W. Allegretto, for taking time to read my thesis and suggest improvements.

I would like to acknowledge the contributions of the Communications Research Centre in fabricating the MSM detector arrays, and the Alberta Microelectronic Centre in fabricating the V-grooves.

A number of people have contributed to the work presented in this thesis. In particular, I wish to thank George Fraser, Graham McKinnon, Marc Veillieux, Shreyas Paranjpe, Michelle Dolman, Trent Cherniwchan, and Rohit Sharma.

Finally, I wish to thank my family, and especially Terri, without whom I undoubtedly would have worked much harder and accomplished much less.

Table of Contents

1 Introduction 1

1.1 Transversal Filters 1

1.1.1 Linear Time-Invariant System Synthesis 2

1.1.2 Matched Filtering 3

1.1.3 Equalization 4

1.1.4 Frequency Filtering 5

1.2 Transversal Filter Implementations 6

1.2.1 Digital Computers 6

1.2.2 Charge-Coupled Devices 6

1.2.3 Switched-Capacitor Devices 8

1.2.4 Surface-Acoustic-Wave Devices 8

1.2.5 Acoustic Charge Transport Devices 10

1.2.6 Superconductive Delay Line Devices 11

1.2.7 Optical Fiber Delay Line Devices 12

1.3 Optoelectronic Weight Setting 14

1.4 Organization of Thesis 16

2 The Optoelectronic Transversal Filter 18

2.1 The Transmitter Module 19

2.2 The Optical Distribution Module 20

2.3 The Receiver Module 24

2.3.1 The Photodetector Array and Fiber Alignment 24

2.3.2 The Receiver Circuit 27

2.3.3 The Wideband Post-Detection Amplifier 28

3 The Multi-Channel Analog Voltage Controller 32

3.1 User Interface 33

3.2 68HC11 Microcontroller and Interface Circuitry 37

3.2.1 IBM-PC to 68HC11 Interface Board 38

3.2.2 Output Interface Board 39

3.2.3 68HC11 Microcontroller Software 40
3.3 Digital-to-Analog Converter 41
3.4 Address Decoding and Sample-and-Hold Circuitry 44

4 System Performance 46
4.1 Optoelectronically Set Weights 46
4.1.1 Variable Bipolar Weights 46
4.1.2 Weight Accuracy 47
4.2 Bandpass Filter Synthesis 49
4.2.1 Tap Weight Determination 50
4.2.2 Relative Importance of Tap Weight and Tap Delay Accuracy 51
4.2.3 Experimental Results 55

5 Summary and Conclusions 60

References 63

Appendix A: Printed Circuit Board Masks and Component Placement Diagrams ... 66

Appendix B: Program Listings 74
B.1 User Interface Software 75
B.1.1 PORTDEFN.H 75
B.1.2 IBM_OETF.C 77
B.2 68HC11 Microcontroller Software 86
B.2.1 intial.doc 86
B.2.2 declare.doc 86
B.2.3 refresh.doc 88
B.2.4 intseq.doc 90
B.2.5 straint.doc 93
B.2.6 updseq.doc 93
B.2.7 reqseq.doc 94
B.2.8 loaddata.doc 95
B.2.9 error.doc 97

Appendix C: Schematic Diagrams.....98

Appendix D: Multi-Channel Analog Voltage Controller Operating Instructions 105

Appendix E: Optoelectronic Transversal Filter 107

List of Tables

Table		Page
3.1.	Parallel port pin assignments	34
3.2.	User interface options	36
3.3.	68HC11 assembly language files	40
4.1.	Bandpass filter Fourier coefficients with Gaussian window	51

List of Figures

Figure		Page
1.1.	Standard transversal filter architecture	1
1.2.	Charge-coupled device delay line	6
1.3.	Pipe-organ transversal filter	7
2.1.	The optoelectronic transversal filter	18
2.2.	Transmitter circuitry	19
2.3.	Silicon V-groove chip with optical fibers	22
2.4.	MSM photodetector array	24
2.5.	Fiber array clamp	25
2.6.	Uniformity measurement	26
2.7.	The receiver circuit	27
2.8.	Amplifier gain stages	28
2.9.	Amplifier frequency response	30
3.1.	Multi-channel analog voltage controller	32
3.2.	IBM-PC to 68HC11 interface board	38
3.3.	Output interface board	39
3.4.	Sallen and Key low-pass noninverting amplifier section	42
4.1.	Transversal filter output as a function of one bias voltage	46
4.2.	MSM photodetector frequency response for various bias voltages	48
4.3.	Ideal periodic bandpass filter function	50
4.4.	Calculated power spectrum of synthesized bandpass filter centered at 200 MHz using a rectangular window function	51

4.5. Calculated power spectrum of synthesized bandpass filter centered at 200 MHz using a Gaussian window function 51

4.6. Calculated power spectrum of synthesized bandpass filter centered at 200 MHz with up to a 10% error in the weights 53

4.7. Calculated power spectrum of synthesized bandpass filter centered at 200 MHz with up to a 25% error in the weights 53

4.8. Calculated power spectrum of synthesized bandpass filter centered at 200 MHz with up to a 10% error in the tap delays 54

4.9. Calculated power spectrum of synthesized bandpass filter centered at 200 MHz with up to a 25% error in the tap delays 54

4.10. Measured power responses of synthesized bandpass filters centered at 200 MHz (solid line) and 350 MHz (dashed line) 55

4.11. Measured phase response of synthesized bandpass filter centered at 350 MHz 56

4.12. Measured power response of synthesized bandpass filter centered at 350 MHz 57

4.13. System response to a 330 MHz sine wave during filter reconfiguration ... 58

A.1 (a) Optoelectronic transversal filter receiver: Board 1 mask 67

A.1 (b) Optoelectronic transversal filter receiver: Board 2 mask 67

A.2 (a) D/A converter board: Component placement diagram - top side 68

A.2 (b) D/A converter board: Component placement diagram - bottom side 69

A.3 (a) D/A converter board: Printed circuit board mask - top side 70

A.3 (b) D/A converter board: Printed circuit board mask - bottom side 70

A.4. Address decoding and sample-and-hold circuit board: component placement diagram 71

A.5 (a) Address decoding and sample-and-hold circuit board: Printed circuit board mask - top side 72

A.5 (b) Address decoding and sample-and-hold circuit board: Printed circuit board mask - bottom side 73

C.1. IBM-PC to 68HC11 interface board 99

C.2. Output interface board 100

C.3. D/A converter board 101

C.4 (a) Address decoding and sample-and-hold circuit board 102

C.4 (b) Address decoding and sample-and-hold circuit board 103

C.4 (c) Address decoding and sample-and-hold circuit board 104

E.1. Optoelectronic transversal filter 112

E.2. MSM sensitivity as a function of dc bias voltage 112

E.3. MSM frequency response for various bias voltages 113

E.4. – Bandpass filter synthesis with center frequencies of 200 MHz (solid line) and 350 MHz (dashed line) 114

List of Symbols

a_m	m^{th} equalizer tap weight
α	Gaussian window parameter
A_v	noninverting amplifier gain
B	bandpass filter bandwidth
C	capacitor
D_n	n^{th} detector
$\delta(t)$	Dirac function
$\Delta\tau$	incremental delay
f	frequency
f_c	bandpass filter center frequency
g(t)	linear time-invariant system impulse response
h(t)	impulse response
H(f)	frequency response
H(s)	transfer function
k	equalizer output index
m	adjusted tap index
M	adjusted number of taps
n	tap index
N	number of taps
R	resistor
t	time
T	incremental delay
T_n	delay of n^{th} tap
V	voltage
V_n	n^{th} bias voltage
w_n	weight of n^{th} tap
ω_0	low pass filter 3 dB cutoff frequency
x(t)	linear time-invariant system input
x_k	n^{th} input to equalizer

$y(t)$	linear time-invariant system output
y_k	n^{th} output of equalizer

List of Acronyms and Abbreviations

ac	alternating current
A	ampere
ACT	acoustic charge transport
Au	gold
cm	centimeter
CCD	charge-coupled device
CMOS	complementary metal-oxide-semiconductor
dB	decibel
dBm	decibels referred to milliwatts
dc	direct current
D/A	digital-to-analog
EMI	electromagnetic interference
FET	field effect transistor
GaAs	gallium arsenide
Ge	germanium
GHz	gigahertz
HACT	hetereojunction acoustic charge transport
Hz	hertz
IBM	International Business Machines
IC	integrated circuit
ISI	intersymbol interference
J	joule
JFET	junction field effect transitor
kΩ	kilo-ohm
K	Kelvin
LSI	large scale integration
m	meter
mA	milliampere
mm	millimeter

ms	millisecond
mV	millivolt
mW	milliwatt
µC	microcontroller
µF	microfarad
µm	micrometer
µs	microsecond
MHz	megahertz
MOS	metal-oxide-semiconductor
MSM	metal-semiconductor-metal
nF	nanofarad
nm	nanometer
ns	nanosecond
Ni	nickel
O	oxygen
Op-Amp	operational amplifier
pF	picofarad
PC	personal computer
rf	radio frequency
s	second
SAW	surface-acoustic-wave
Si	silicon
V	volt
Ω	ohm
W	Watt

Chapter 1

Introduction

This chapter will describe the concept of a transversal filter and briefly summarize a few of its signal processing applications. Existing implementations will then be reviewed in detail to show how the present work extends the state of the art. The concept of setting analog weights by optoelectronic means will also be introduced, followed by a description of the present work, and the organization of this thesis.

1.1 Transversal Filters

Transversal filters were first described in 1940 by Kallmann [1] as "the electrical analogue to the grating spectroscope" and have also come to be known as tapped-delay-line filters. The basic elements of an N-tap transversal filter are shown in Figure 1.1. The output

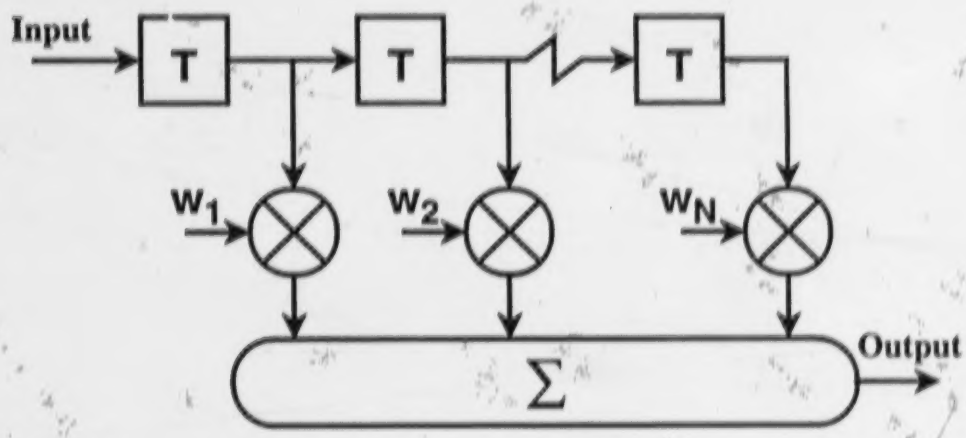

Figure 1.1. *Standard transversal filter architecture*

is formed by additively combining the N weighted and delayed copies of the input. The impulse response $h(t)$ is thus given by

$$h(t) = \sum_{n=1}^{N} w_n \delta(t - nT) \tag{1.1}$$

where the w_n are the weights, T is the delay between adjacent taps, and $\delta(t)$ is the Dirac function. The corresponding transfer function is given by

$$H(f) = \sum_{n=1}^{N} w_n e^{-j2\pi nTf}. \tag{1.2}$$

Although in general the weights are complex and the delays are arbitrarily spaced, applications often require real weights and uniformly spaced delays. The following sections will describe some of the more common transversal filter applications.

1.1.1 Linear Time-Invariant System Synthesis

The most general transversal filter application is the synthesis of linear time-invariant systems. In the time domain, this process can be thought of as a sampling of the desired system impulse response. Consider a causal system whose impulse response is $g(t)$ with a causal input signal $x(t)$. The output $y(t)$ of this system is defined by the convolution integral

$$y(t) = \int_0^t x(\tau) g(t-\tau)\, d\tau = \int_0^t x(t-\tau) g(\tau)\, d\tau \tag{1.3}$$

which may be approximated by

$$y(t) \approx \sum_{n=0}^{t/(\Delta\tau)} x(t - n\Delta\tau) g(n\Delta\tau) \Delta\tau. \tag{1.4}$$

This approximation is precisely the response of an N-tap transversal filter where $N = t/(\Delta\tau) + 1$, $T = \Delta\tau$, and $w_n = g(n\Delta\tau)\Delta\tau$.

In the frequency domain, a transversal filter approximation to a linear time-invariant system can be designed by putting the transversal filter transfer function in the form of a truncated Fourier series expansion of the desired transfer function. Assuming uniform delay increments of T, the transversal filter transfer function is given by

$$H(f) = \sum_{n=1}^{N} w_n e^{-j2\pi nTf}. \tag{1.5}$$

To put this equation in the form of the complex exponential Fourier series it is necessary to define an index $m = n - (N+1)/2$ such that m runs from $-M$ to M where $M = (N-1)/2$. Note that this requires N to be odd. This substitution yields

$$H(f) = e^{-j(N+1)(\pi T)f} \sum_{m=-M}^{M} w_m e^{-jm(2\pi T)f} \tag{1.6}$$

The redefinition of the index is consistent with the constraint of causality: it produces a linear phase term in the frequency response which in the time domain corresponds to a time delay of one-half the sampled impulse response. This effectively shifts the sampled impulse response into the causal ($t \geq 0$) region. The linear time-invariant system may then be synthesized using standard Fourier series design techniques. Since the series is truncated, any discontinuities which may be present in the function which is being approximated will give rise to Gibbs phenomenom ripples. Thus it may be desirable to modify the Fourier coefficients w_m with a suitable window function. Note that the system frequency response will be periodic in the frequency domain with period $1/T$. If this periodicity is undesirable, passive electrical components may be used to limit the frequency response to one period. Also note that the incremental delay determines the period of the frequency response; a wider system bandwidth will be of limited use if the incremental delay cannot be made small enough to include it in one period of the synthesized response.

While the synthesis of linear time-invariant systems will in general require complex weights, it is often more practical to realize transversal filters which implement real weights. With complex weights, systems may be designed which independently process the in-phase and quadrature components of a signal; with real weights both signal components must be processed by the same transfer function. An equivalent statement is that systems with complex weights may process signals on carriers while systems with real weights process baseband signals. It is possible to synthesize a transversal filter with complex weights by using two systems with real weights to process the in-phase and quadrature signal components separately.

1.1.2 Matched Filtering

A matched filter is a linear time-invariant system which yields the maximum possible output in response to a specific finite time message signal in the presence of additive white noise. It can be shown [2] that the impulse response of this optimum system is a time-reversed replica of the message signal. As was shown in the previous section a transversal filter may be used to approximate the desired impulse response; the complexity of the mes-

sage signal will determine the minimum number of taps necessary to form an adequate approximation.

Spread spectrum communication systems often use the matched filtering concept. For example, the technique of code division multiplexing involves the simultaneous transmission of multiple channels by defining orthogonal multiple bit binary codes for each channel - these codes are then combined and transmitted to receivers which consist of filters matched to one of the channel codes. Transversal filter architectures may be used both in the encoding transmitter and the decoding receivers.

1.1.3 Equalization

In digital communication systems, a significant source of signal degradation is often the dispersion of the transmission channel, which causes intersymbol interference (ISI). Particularly prone to this type of degradation are telephone and digital mobile radio channels. When the dispersion of the message signal is over a moderate number of symbol intervals, ISI may be drastically reduced by using a transversal filter configured as a zero-forcing equalizer. If the channel dispersion characteristic is known, the tap weights of a $2M+1$ tap transversal filter can be chosen so that each dispersed symbol has a peak value at the correct sample time and is zero for M sample times before and after the correct time. Consider the output y_k of a $2M+1$ tap transversal filter with input x_k and tap weights a_m at sample time k:

$$y_k = \sum_{m=-M}^{M} a_m x_{k-m} \qquad ;k = 0, \pm 1, \pm 2, \ldots, \pm M. \tag{1.7}$$

To minimize ISI we also require

$$y_k = \begin{cases} 1 & k = 0 \\ 0 & k = \pm 1, \pm 2, \ldots, \pm M \end{cases} \tag{1.8}$$

where the peak value at the correct sample time has been normalized to one for convenience. Combining equations (1.7) and (1.8) yields a set of $2M+1$ simultaneous equations which can be solved for the a_m. Note that this type of equalizer does not completely eliminate ISI; it may in fact introduce some ISI outside the range of $\pm M$ sample points at which it attempts to introduce nulls. Also note that the finite precision with which the tap weights can be set will in general degrade the quality of the nulls.

When the channel dispersion characteristic is not known, or varies with time, an alternate means of determining the values of the weights is required. Preset equalizers iteratively adjust the tap weights based on a series of test symbols which are sent prior to or during pauses in data transmission. Adaptive equalizers iteratively adjust the tap weights by using the data signal itself. Note that adaptive equalizers require the tap weights to be rapidly programmable; transversal filter realizations with static weights are not suitable for this application.

1.1.4 Frequency Filtering

A typical application of the transversal filter architecture in digital signal processing is the design of finite impulse response frequency filters. A typical constraint of such filters, to insure distortionless transmission in the passband, is that they have a linear phase response. Equation (1.6) shows that the transfer function of an analog transversal filter can be written as the product of a linear phase term and a truncated exponential Fourier series. Thus a linear phase response can be ensured by making the Fourier series define a real function; this is accomplished by setting $W_{-m} = \overline{W_m}$ where the overline denotes the complex conjugate.

An interesting (and frequently occurring) subset of linear phase frequency filters are filters formed with real weights; in this case the constraint above reduces to $W_{-m} = W_m$ and the Fourier series component of equation (1.6) may be written as

$$\sum_{m=-M}^{M} W_m e^{-jm(2\pi T)f} \triangleq W_0 + 2\sum_{m=1}^{M} W_m \cos(2\pi mTf) \tag{1.9}$$

since

$$W_{-m}e^{j2\pi mTf} + W_m e^{-j2\pi mTf} = 2W_m \cos(2\pi mTf). \tag{1.10}$$

This implies that the impulse response of the synthesized filter will be an even function in the frequency domain since a sum of even functions is also an even function. Note that this further implies that only one half of the $1/T$ bandwidth can be independently designed; the quantity $1/(2T)$ will hereafter be refered to as the programmable bandwidth. This constraint is essentially the analog equivalent of the Nyquist sampling theorem which states that in order to restore a sampled signal perfectly, it must be sampled at twice the frequency of the highest frequency component present in the signal. In this case the tap spacing must occur at twice the frequency of the programmable bandwidth.

1.2 Transversal Filter Implementations

This section will review the various approaches which have been taken to solve the key transversal filter implementation problems of forming delay lines and weighted taps. The technologies used will be described and state-of-the-art devices as reported in the published literature will be compared in terms of signal bandwidth (or processing speed), number of taps, incremental tap delay, and tap weight resolution and reconfiguration time where applicable.

1.2.1 Digital Computers

One of the simplest approaches to the transversal filter implementation problem is to synthesize the filter on a computer. In this manner filters with a practically unlimited number of taps and tap weight resolution may be realized. The principal drawback is that there is a trade-off between filter complexity and the maximum real time signal processing speed. Since the inherently parallel transversal filter architecture is being synthesized on a serial processor, as the number of taps or the tap weight resolution increases the real time signal processing speed decreases. In addition, the minimum delay between taps may be limited by the speed of the analog-digital conversion process. For these reasons, digital computer based transversal filters are most often used where the required processing speed is low, such as in audio signal processing, or where it is not necessary to process the signals in real time, such as in seismic or radio astronomy data processing. It is interesting to note that in some applications where digital computers could be used, analog filters may be preferred because they introduce savings in size, power, and/or cost.

1.2.2 Charge-Coupled Devices

Charge-coupled devices (CCD's) are analog delay lines in which the input signal is sampled and stored as a charge packet which is then passed down the delay line by a sequence of clock pulses. A schematic representation is shown in Figure 1.2. The delay line typically consists of a series of field effect transistors (FET's) connected drain-to-source with the charge packet stored on a capacitor connected between each drain and gate. The clock pulses alternately drain the charge from each capacitor and store it on the next capacitor. The transversal filter output may be formed by non-destructively sensing the charge at each delay stage, and weighting and summing the results. Devices of this type have been constructed with clock rates of up to 10 MHz and cumulative delays of ~1 second [3], and as many as 1024 stages [4], with both static and electrically programmable weights [3,4].

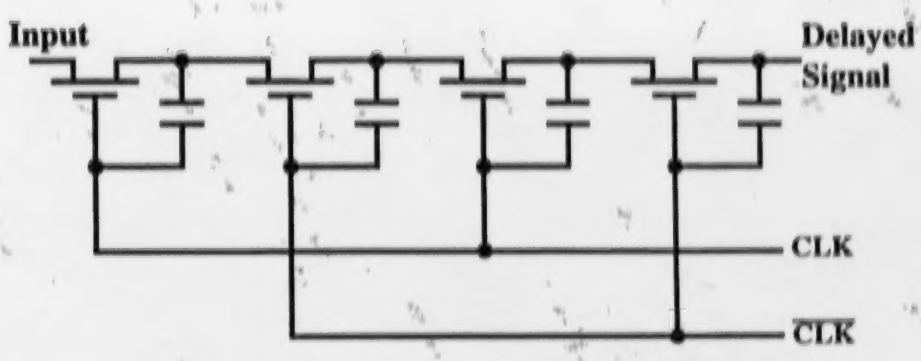

Figure 1.2. *Charge-coupled device delay line*

The processing speed of a conventional CCD transversal filter is limited by the non-CCD electronics necessary to implement the transversal filter function: namely the clock drivers and clock coupling to the tapping electrodes, and the summing and weighting circuitry. A novel CCD transversal filter which partially alleviates these difficulties was demonstrated by Chiang and Burke [5] in 1983. It makes use of the "pipe-organ" structure shown in Figure 1.3; this structure avoids bandwidth limitations due to the tapping circuitry by presenting the input signal to N separate weighting and delay circuits. The weights are

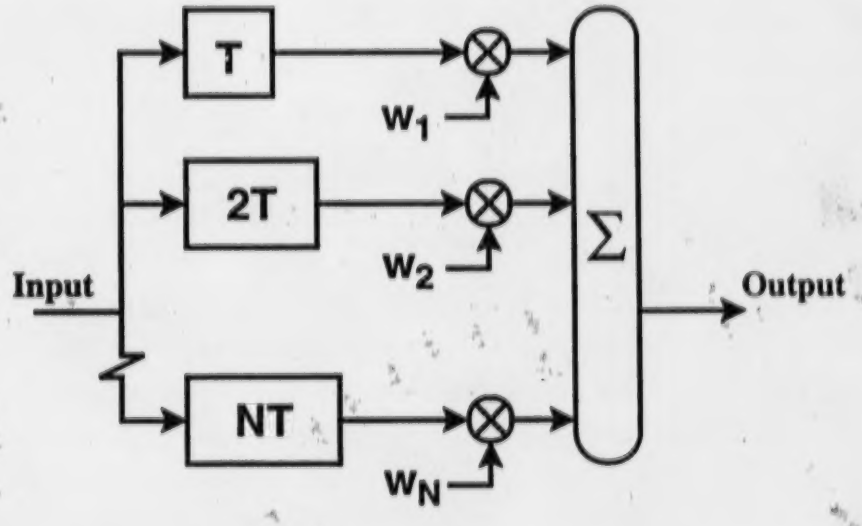

Figure 1.3. *Pipe-organ transversal filter*

applied by 6-bit CCD multiplying digital-analog converters. With this structure a 32-stage transversal filter was demonstrated at a clock rate of 25 MHz with a dynamic range of more than 60 dB.

1.2.3 Switched-Capacitor Devices

Switched-capacitor devices implement signal processing functions using only capacitors, operational amplifiers (Op-Amps), and analog switches. They are typically fabricated in monolithic form using high-component-density MOS technology; transversal filters built with this technology thus have the advantage that most of the support circuitry as well as digital signal processing circuitry can be monolithically integrated with the filter. Both programmable and nonprogrammable filters have been demonstrated. A typical architecture is described by Lee and Martin [6] in which eight 32-stage transversal filters with fixed weights are fabricated on a single CMOS integrated circuit. It consists of 32 sample-and-hold circuits, a 32 X 32 "rotating switch," and eight 32-input switched capacitor summing amplifiers. The input is sampled sequentially by the sample-and-holds and the rotating switch keeps the first input to each summing amplifier connected to the sample-and-hold which contains the most recent sample. The second input is connected to the next most recent sample and so on. The values of the weights are set by the size of the input capacitor of each summing amplifier; the sign of each weight is determined by the node to which the capacitor is connected to in the amplifier. A programmable filter was demonstrated in 1981 by Sunter et. al. [7] which used multiplying digital-analog converters to implement the weights. The primary drawback of these filters is their limited processing speed: the highest demonstrated clock speed is only 2 MHz [6].

1.2.4 Surface-Acoustic-Wave Devices

Surface-acoustic-wave (SAW) devices rely on the piezoelectric effect to convert an electric signal to a relatively slowly propagating acoustic wave, usually on quartz or lithium niobate substrates. The electric fields associated with the propagating acoustic wave may then be non-destructively sampled by capacitive taps and the results weighted and summed to form a transversal filter. The travelling acoustic wave is typically created with a transducer consisting of a thin metallic film on the surface of the piezoelectric substrate; taps are typically formed from thin metallic fingers perpendicular to the direction of propagation of the SAW either on the surface of the piezoelectric material or on the surface of an external integrated circuit separated from the piezoelectric material by a small air gap on the order of a few hundred nanometers. Programmable tap weights are implemented using MOS cir-

cuitry on a separate GaAs or Si integrated circuit which is connected to the piezoelectric material either by air gaps or wire bonds. Note that since the tapped signals actually pass through the weighting circuitry, bandwidth limitations may be imposed depending on the technology used. Bipolar weights are generally implemented by fabricating two sets of taps and weights, each set being connected to a passive summing bus; the two summing busses are then used as inputs to a differential amplifier. With this technology programmable transversal filters with hundreds of taps, incremental delays of several nanoseconds, and signal bandwidths of one to two hundred megahertz have been demonstrated.

The primary limitations of these devices are the limited bandwidths of the SAW transducers and the weighting circuitry. Even if transducers with 3 dB bandwidths of 500 MHz could be realized it would be difficult to fabricate a transversal filter with that much programmable bandwidth. It would require a tap sampling rate of 1 GHz (to satisfy the Nyquist criterion) which corresponds to an incremental delay of one nanosecond; since the speed of sound in lithium niobate is ~3500 m/s this implies a center-to-center tap spacing of 3.5 μm. At such high frequencies electromagnetic coupling between the tapping fingers may introduce additional problems. The following paragraphs will give examples of state-of-the-art SAW programmable transversal filters.

Researchers at the Massachusetts Institute of Technology's Lincoln Laboratory have developed a SAW/FET transversal filter with 350 programmable taps spaced 15 μm apart to give a programmable bandwidth of 116 MHz [8,9]. To drive the lithium niobate SAW delay line they utilize an edge-bonded transducer which has a bandwidth of 100 MHz centered at 175 MHz. The taps consist of coupling fingers fabricated on a silicon integrated circuit; the silicon chip is pressed against the lithium niobate delay line with an air gap of 300 nm maintained by SiO_2 rails deposited on the chip. In a 1986 version [8], the weights are formed by a MOS variable capacitive divider circuit. This is essentially a sample-and-hold circuit with a MOSFET associated with each tap allowing the transfer and storage of an analog programming voltage on a MOS variable capacitor. In a 1988 version [9], the electrically variable capacitors were replaced with electrically variable resistors. These are MOSFET's whose source-to-drain resistance is controlled by the dc voltage stored on the capacitance of the gate. This later version exhibited a tap on/off ratio of greater than 30 dB and could be reconfigured in 30 μs. It should be noted, however, that this reconfiguration speed depends on the availability of an appropriate time-varying analog programming waveform; arbitrary reconfiguration was not demonstrated. These devices were demonstrated as adaptive, matched, and narrow-bandwidth filters.

Researchers at Texas Instruments have developed a "Digitally Controlled Programmable Transversal Filter" with 32 taps spaced 14.5 μm apart to give a programmable bandwidth of 120 MHz [10]. A transducer with a bandwidth of 100 MHz centered at 300 MHz drives a lithium niobate SAW delay line. The taps consist of quarter wavelength (at 240 MHz) active electrodes interleaved with ground electrodes fabricated on the piezoelectric substrate. The tap weights are formed by digitally controlled variable gain amplifiers fabricated on a separate LSI GaAs IC; the taps are connected to the weight circuits by wire bonds. The amplifiers consist of segmented dual-gate FET pairs where the gain of each FET is binary-scaled; this produces bipolar weights with 7 bits of resolution which can be arbitrarily reconfigured in less than 100 μs. The device was demonstrated as bandpass filter with programmable center frequency and bandwidth.

1.2.5 Acoustic Charge Transport Devices

Acoustic charge transport (ACT) technology is essentially a combination of CCD and SAW technologies. A single frequency SAW is used to produce travelling potential wells in a nearby depleted channel in a semiconductor material; in this manner charge packets may be transported at the speed of sound in the semiconductor. The charge packets may be non-destructively sensed via capacitive coupling to electrodes. The tap weighting techniques used in SAW filters can also be applied to ACT filters. Possibly the most attractive feature of ACT technology is that GaAs may be used as the piezoelectric substrate. Normally GaAs is unsuitable because of its low piezoelectric coupling coefficient. In ACT devices, however, this difficulty may be overcome by using a long narrow-band transducer since only a single frequency is used to generate the acoustic wave. This allows the fabrication of completely integrated programmable transversal filters; all the tapping, weighting, summing, amplifying, and programming circuitry may be realized on the same substrate as the transducer and propagating SAW. The ultimate limitation of this technology, however, is the same shortcoming as seen in SAW transversal filters: the programming bandwidth is limited by the minimum achievable proximity of the tap weights. In GaAs, this problem is worsened by the fact that the speed of sound in GaAs is only 2860 m/s. Thus a sampling rate of 1 GHz would require a center-to-center tap spacing of 2.86 μm. While a pipe-organ structure could conceivably allow the fabrication of devices with smaller incremental delays, such a structure has its own problems and there has yet to be a demonstration of this type of device.

An ACT based programmable transversal filter has been developed as a commercial product by Electronic Decisions Incorporated. A full page ad in an early 1991 issue of the

IEEE Spectrum descibes a programmable transversal filter with 128 taps and a 150 MHz signal bandwidth. Earlier versions of this device in the published literature include a 32-tap programmable transversal filter with a programmable bandwidth of 45 MHz and 8 bits of tap weight control [11]. Tap weights were implemented by controlling the ac impedance of forward biased GaAs schottky barrier diodes via two complementary current sources; there are separate diode arrays for positive and negative weights. A more recent paper describes a 64-tap programmable transversal filter with a tap spacing of 22 μm to give a programmable bandwidth of 65 MHz [12]. The input transducer is driven with a 360 MHz sine wave; this corresponds to an input sampling rate of 360 MHz and a signal processing bandwidth of 180 MHz. Tap weights were implemented with 6 bits of accuracy using multiplying digital-analog converters monolithically integrated with the delay line.

A heterojunction ACT (HACT) transversal filter, a slight variation of ACT technology, has been developed at the United Technologies Research Center [13]. The principal innovation is the use of potential steps resulting from band structure discontinuities between heterostructure layers to achieve vertical charge confinement. Conventional ACT devices rely on the application of dc potentials on the top and bottom surfaces of the GaAs wafer to achieve electrostatic charge confinement. The advantages of a HACT device are primarily in an improvement in the coupling to the non-destructive sense electrodes and in an increase in the SAW potential in the transport channel; these improvements are possible because the HACT device is a thin film structure and thus the distance between the electrodes, the channel, and the SAW may be minimized. A transversal filter with 480 taps, input sampling rate of 144 MHz, programmable bandwidth of 72 MHz, and limited programmability was demonstrated in 1989 [14]. The equally weighted taps were alternately connected to the positive and negative inputs of a differential amplifier.

1.2.6 Superconductive Delay Line Devices

A possible candidate for signal processing in the 2-20 GHz range is the superconductive delay line transversal filter [15]. While superconductive delay lines may carry wideband signals for reasonably long delays with very low dispersion, and the velocity of propagation along the delay line allows for sufficiently low incremental delays between taps to give very wide programmable bandwidths, to date tapping schemes have been plagued by unwanted reflections, undesirable coupling paths, and other post-design features [16] which have prevented the demonstration of a transversal filter.

1.2.7 Optical Fiber Delay Line Devices

Optical fiber is an excellent delay medium. Over the maximum distances typically required in transversal filters (~10 m), even the dispersion of multimode fiber will not cause distortion in the programmable bandwidth of the filter. Since the velocity of propagation of the signal carrier is the speed of light in glass, ~2 x 10^8 m/s, very short incremental tap delays, leading to very large programmable bandwidths, may be implemented. For example, a programmable bandwidth of 5 GHz, requiring incremental tap delays of 0.1 ns, corresponds to an center-to-center tap spacing of 2 cm. Another useful property of optical fiber is its immunity to electromagnetic interference. This eliminates many of the cross-coupling difficulties which limit the signal bandwidth in other technologies. The principal design difficulties involved in the construction of transversal filters are in finding means of implementing the taps and the weights. To date, the only convincing demonstrations of programmable weights have utilized variable optical couplers [22,27] and piezoelectric modulators [22]; presumably due to the expensive nature of these devices only 3-tap filters were constructed. The principal bandwidth limitations on optical fiber transversal filters are imposed by the optoelectronic conversion devices; lasers and photodetectors with bandwidths of a few gigahertz are already readily available and detectors and external modulators with bandwidths of up to 100 GHz are currently under development and are becoming available. In purely optical computing systems this will not be a limitation but such systems will probably not be commercially available or needed in the near future.

A number of attempts have been made to exploit the signal processing capabilities of optical fiber transversal filters [17-27]. The first demonstration, made by Chang *et al.* in 1977 [17], was of a 15-fiber filter with approximately uniform static tap weights with taps spaced 1.05 m apart giving a fundamental frequency of 193 MHz. The pipe-organ architecture was used and the taps were achieved by imaging the modulated light source on a bundle of fibers held together inside a hypodermic needle. A similar arrangement was used to image the light emerging from the other ends of the fibers on a single large photodetector. A 19-tap filter using the standard transversal architecture was demonstrated in 1982 by Jackson *et al.* [18]. A single fiber wrapped around a cylinder was tapped by forming microbends in the fiber with a tapping pin. The emerging light was then focussed onto a single detector. In this manner very uniformly spaced and weighted taps were realized with a spacing of 20 cm to give a fundumental frequency of 1 GHz. A later version of this device implemented static weights of 0 or 1 by introducing a mask between the collecting and summing optics [19]. Another device which relied on a mechanical system of forming the taps was published by Newton *et al.* in 1983 [20]. It too consisted of a single fiber looped around a cyl-

inder but the taps were formed by positioning the fibers in V-grooves etched in silicon and removing a portion of the cladding of each fiber by polishing flat the surface of the V-groove substrate. By mating this device with a similar device with individual output fibers, four evanescently coupled taps spaced 30 cm apart were realized. The output fibers could then be masked before being coupled to a single detector. A problem common to all these devices is that since high-speed photodetectors have small sensitive areas to minimize device capacitance, imaging optics are required to gather the light from the fiber taps and focus it on the photodetector. As larger numbers of taps are used a greater demagnification is required which forces the imaging lens to be placed further away from the taps and this in turn reduces the light collection efficiency [19]. In addition, mechanical stability is difficult to achieve with such a lens system.

A number of optical fiber transversal filters have been demonstrated which do not suffer from the imaging lens problem. In 1987, Pappert *et al.* [23] described an 8-tap transversal filter with uniform weights for rf direction finding. This was essentially a phased array radar system which used optical fiber to implement the necessary delays. The microwave antennas of the array each modulated the current to a laser diode whose optical output was coupled into the optical fiber taps. Optical summation was achieved with an 8 X 1 fiber coupler; note that this introduces an excess loss of 10 log (1/8) = 9 dB in optical power since the 8 X 1 coupler used is really an 8 X 8 coupler with seven outputs missing. Also in 1987, Wang [24] demonstrated a 256-tap transversal filter with a similar optical summation technique. It featured approximately uniform fixed weights and an incremental delay of 0.308 ns which corresponds to a fundamental frequency of 3.25 GHz. This large number of taps was achieved by cascading eight 2 X2 fiber couplers. An alternative means of implementing large number of taps with fixed weights was demonstrated in 1987 by Lee *et al.* [25]. Their filter consisted of a single length of fiber which contained a series of twenty dielectric mirrors 30 cm apart produced by a novel fusion-splicing technique; uniform weights were achieved. A 1989 paper from the same research group [26] described a very similar filter with 16 taps 10.2 cm apart with uniform weights to within ±10 percent. In 1990, Gookin and Berry [27] made the only really convincing demonstration to date of an optical fiber transversal filter with programmable analog weights. It consisted of a 3-tap filter with incremental delays of 10 cm using the pipe-organ architecture with optical taps and summation realized with 1 X 4 optical fiber star couplers and tap weights set with Mach-Zender integrated optic couplers. The cost of the variable fiber couplers render filters of this type impractical for use with a larger number of taps.

The technology of coherent optical processing has been applied to the transversal filter implementation problem. The first demonstration of a coherent optical fiber transveral filter, in which the amplitude and phase of the light carrier represents the signal, was made by Davies and James in 1984 [21]. It consisted of a delay line with two taps formed by piezoelectric phase modulators spaced 452 m apart. The signal to be processed was first put on an electrical carrier and then delivered simultaneously to the two phase modulators. The authors proposed that complex weights could be implemented by weighting the electrical signal in both magnitude and phase before driving the phase modulators but did not demonstrate this. In 1986, Jackson *et al.* [22], reported operation of a coherent optical fiber delay-line processor with three taps which was an all-fiber circuit except for the input to the acousto-optic modulator and the output from the photodetector. Taps and the amplitudes of the weights were realized with variable directional couplers and the phases of the weights were controlled by varying the dc voltage applied to a piezoelectric ring around which the fiber was wound. Only on/off amplitudes and 180 degree phase shifts were demonstrated for tap variability. Coherent optical fiber transversal filters have not been developed further probably due to the cost of the variable directional couplers and the piezoelectric phase modulators, the need for a long coherence length optical source, the extreme sensitivity of the device to environmental fluctuations, the need for polarization controllers to align the polarizations of the interfering signals, and the restriction to incremental delays of least ~10 ns by the geometry and characteristics of the piezoelectric modulators.

Recently, integrated optics technology has been used to implement coherent optical transversal filters [35 - 36]. Silica-based single mode waveguides fabricated on a three inch silicon wafer were used to make a four-tap transversal filter with an incremental tap delay of 4 cm [35], and an eight-tap transversal filter with an incremental tap delay of 1 cm [36]. Complex weights were implemented using tunable splitters consisting of a Mach-Zehnder modulator with a thin film heater on one arm, and phase shifters consisting of a thin film heater on a waveguide. The thin film heaters were used to shift the phase of the optical carrier in the underlying waveguides by the thermo-optic effect. The principal limitation on devices using integrated optics technology is the maximum available delay length; this restricts the implementations to filters with small numbers of taps and very wide programmable bandwidths.

1.3 Optoelectronic Weight Setting

An intensity modulated optical signal may be weighted as it is converted to an electrical signal by using a photodetector whose responsivity may be varied. In general, there are

three characteristics required of photodetectors which are to be used to apply weights optoelectronically: (1) the photodetector's frequency response should be flat over the frequencies of interest so that the application of a weight does not distort the signal, (2) there should exist some means by which the effective responsivity of the detector may be varied, and (3) the shape of the frequency response should be constant over all weight values. The first requirement is not very strict: in many cases the frequency response exhibits a smooth and continuous distortion which may be equalized with acceptable loss using passive components. While it is not necessary, it is often desirable that the photodetector responsivity be a simple function (linear or quadratic, for example) of some control parameter, such as the bias voltage. This simplifies calibration of the device. The requirement regarding the shape of the frequency response may be problematic for some types of photodetectors. For example, changing the value of the control parameter may also effect the speed of the device; the net effect of this is to reduce the accuracy with which weights may be applied.

There have recently been several examples in which analog weights were applied to intensity modulated optical signals by controlling the effective responsivity of the photodetective receiver circuit [28 - 30].

Yonezu *et al.* [28] demonstrated an optoelectronic neural network in which optical synaptic inputs were weighted with current mirror multiplier circuits after detection by photodiodes. In this manner an electrical bias could be used to control the effective photocurrent.

MacDonald and Lee [29] proposed an optoelectronic neural network in which the weights could be set by directly controlling the photoresponse of GaAs interdigital photoconductors by modulating the bias voltages across the detectors. These detectors are characterized by their photoconductive gain (or attenuation) which is the ratio of the photocarrier transit time to the carrier lifetime. Since the transit time depends on the carrier drift velocity which in turn depends on the applied bias voltage, the photoconductive gain may be controlled by the bias. There are several advantages to this approach. First, processors which apply weights in this manner can have very wide bandwidths since these interdigital detectors can be fabricated with very low device capacitance. Second, the weights are determined by a set of voltages with magnitudes typically less than 5 V. Such voltages are easily generated and can be switched rapidly with widely available electronic components. Third, the bidirectional nature of these devices allows the implementation of bipolar weights; reversing the polarity of the bias results in a 180° phase shift in the output signal. Other techniques for weighting intensity modulated optical signals, notably those that rely

on variable optical attenuators such as variable optical couplers or spatial light modulators, require twice as many weights to achieve bipolar operation. Typically, there is a set of "positive" weights and a set of "negative" weights each of whose light is summed optically and detected before being subtracted electrically. Such an arrangement also requires twice as many control signals and hence will take twice as long to reconfigure. Fourth, these detectors exhibit a very large on/off ratio, 50-70 dB, which allows for very wide dynamic range weight setting.

A very similar approach to setting weights which uses GaAs interdigital metal-semiconductor-metal (MSM) photodetectors was described by Nitta *et al.* [30]. They too used the bias voltage to control the photocarrier drift velocity and hence the photocurrent. In so doing they found that over most of the bias range (-8 V to +8 V) a linear relationship exists between the photocurrent generated in their device and the bias voltage. A problem inherent in both MSM diodes and photoconductors is that the photodetector's speed is related to the photocarrier's drift velocity; reducing the bias voltage will also reduce the detector's speed and as a result high frequencies will receive smaller relative weights than low frequencies. Because of this problem, either a reduced bandwidth or lower weight accuracy must be accepted.

1.4 Organization of Thesis

The purpose of this thesis is to demonstrate a 16-tap optical fiber transversal filter with a 500 MHz programmable bandwidth and optoelectronically set bipolar weights. To facilitate calibration of the device and to allow demonstration of an agile filter application, a multi-channel analog voltage controller with a personal computer interface has also been designed, constructed, and tested.

The optoelectronic filter work described in this thesis forms a part of the Telecommunications Research Laboratories programme on optoelectronic broadband switching and signal processing carried out by R. I. MacDonald. This programme has been selected as an activity of the Canadian Institute for Telecommunications Research (CITR), one of the National Networks of Excellence, and forms CITR Project A.2.3.

Chapter 2 describes the design of the optoelectronic transversal filter. The construction, testing, and performance specifications of the transmitter, optical distribution, and receiver modules are discussed.

Chapter 3 decribes the design of the multi-channel analog voltage controller. The user interface, microcontroller, digital-analog converter, and address decoding and sample-and-hold modules are discussed. The performance specifications of the controller are given.

Experimental results are presented in Chapter 4. The performance of the filter as an agile bandpass filter is given. Sources of performance degradation are analyzed and suggestions for improvement offered.

A summary of the work done for this thesis and conclusions reached are given in Chapter 5. Suggestions for future work are offered.

Chapter 2

The Optoelectronic Transversal Filter

This chapter will describe the design and construction of the optoelectronic transversal filter. The performance characteristics of the various modules composing the system will be given and discussed. An overview of the system is shown schematically in Figure 2.1.

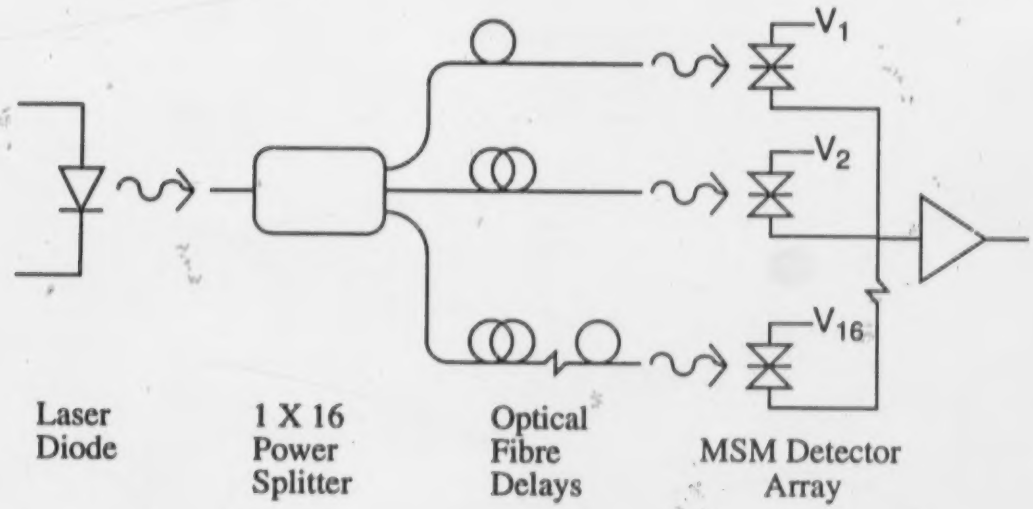

Figure 2.1. *The optoelectronic transversal filter*

The transmitter module consists of a fiber-pigtailed, short wavelength laser diode package with some driver circuitry and mechanical packaging. The optical distribution module consists of a 1 X 16 fiber power splitter, to which optical fiber delays are fusion spliced, and a V-groove fiber alignment system. The receiver module consists of the MSM photodetector array chip, and a four-stage wideband amplifier in a custom built mechanical package. The following sections will describe each module, and the techniques used in their construction, in more detail.

2.1 The Transmitter Module

The rf electrical signal input to the transversal filter is converted to an intensity modulated optical signal by the transmitter module. The optical signal is generated by a Mitsubishi ML6411C model number PT-150 short wavelength (776 nm) multimode fiber (50/125 μm graded index) pigtailed laser diode whose optical output power is a linear function of its bias current for currents above the threshold value for lasing. The laser is maintained at an operating point in the middle of its output power range by a constant current source. According to the manufacturer's specifications the operating point bias current of 65 mA should produce an optical power at the end of the fiber pigtail of 4.765 mW but apparently due to aging effects the optical power measured during assembly of the transmitter module was somewhat less: ~3.0 mW; this was still sufficient, however, for the demonstration of the transversal filter. The rf input signal directly modulates the dc bias current to produce the intensity modulated optical output signal. The circuitry used to maintain the constant bias current and impedance match and ac couple the electrical input signal is shown in Figure 2.2. Since the dynamic resistance of the laser diode when it is lasing is ~3 Ω, a 47 Ω resistor is used to match the laser to the rf source impedance of 50 Ω.

The constant current source consists of an LM317 adjustable voltage regulator with an essentially constant impedance load. The load is formed from four resistors in parallel to

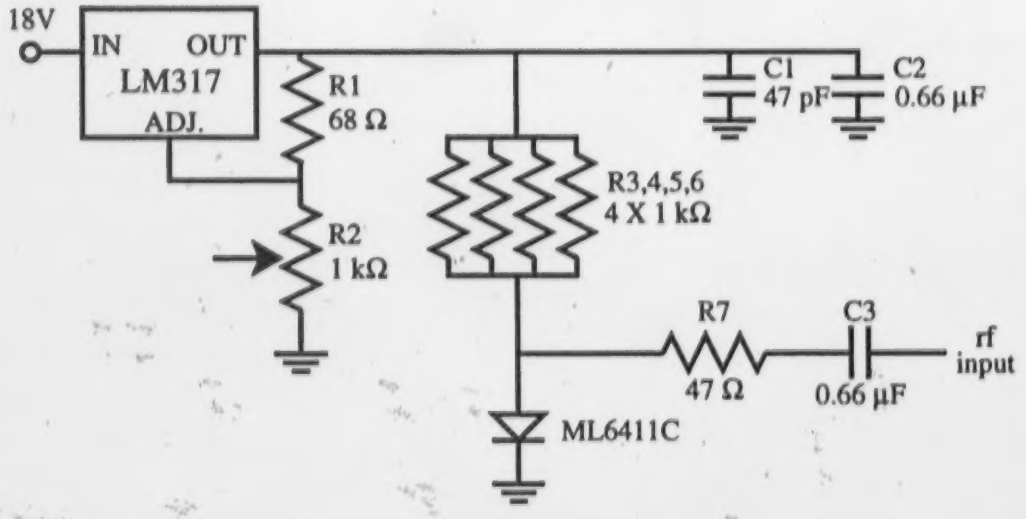

Figure 2.2. *Transmitter circuitry*

meet their maximum power specification of 0.25 W. The load also serves to block the rf input signal from entering the regulator circuitry since the rf signal will see the 250 Ω load in parallel with the 3 Ω resistance of the laser. Capacitors C1 and C2 shunt the attenuated rf input to ground and decouple regulator fluctuations. Two capacitors of different types are used to obtain a suitable response over a large frequency range; the larger ceramic capacitor is effective at frequencies below a few hundred megahertz and the smaller chip capacitor is effective up to ~ 1 GHz.

The back facet monitoring photodetector which is incorporated in the laser diode package was not used to stabilize the output power of the laser as is typically done. Stabilization was omitted here for reasons of component availability and time constraints. It was found that after several minutes of operation thermal equilibrium was reached and thermal drift of the laser's output power was tolerable.

A more serious difficulty encountered with this transmitter was the sporadic mode hopping of the laser diode. This resulted in the output power suddenly shifting ~2 dB. This effect could be replicated by adjusting the bias current through the laser. A feedback loop to stabilize the bias current would not help, however, since the author has observed the same behavior in transmitters using such a circuit. While the details of this behavior are not well understood, it is believed that the mode hopping is primarily due to reflections from fiber components and the fiber ends back into the laser diode since the introduction of an optical isolator between the laser diode and the output fiber has solved such problems in other systems [31]. Unfortunately, a short wavelength optical isolator was not available to test such a solution.

2.2 The Optical Distribution Module

The intensity modulated optical signal produced by the transmitter is split sixteen ways, delayed, and prepared for coupling to a photodetector array by the optical distribution module. The splitting is accomplished by a 1 X 16 multimode fiber power splitter the input of which is fusion spliced to the fiber pigtail of the transmitter's laser diode package. The optical power of the sixteen outputs is specified by the manufacturer to be uniform to within ±0.5 dB; this figure was confirmed by measurements performed during the assembly of the module.

A difficulty encountered with this device is that the splitting ratios appear to be dependent on the laser's output mode structure. The intensity of the light emitted by some of the output fibers relative to other output fibers was actually observed to change when the laser

hopped from one mode structure to another. The sensitivity of the splitting ratios to the mode profile of the splitter's input fiber was also demonstrated by moving a section of the input fiber; this changed the mode profile input to the splitter and the optical power from one of the output fibers was observed to change. This particular problem made accurate calibration of the transversal filter very difficult, especially since the mode hopping of the transmitter is apparently a random process.

Alteration in the power division ratio affects the calibration process because each tap weight is set by comparing the output level of the system, with all tap weights equal to zero except for the tap weight which is being set, with the output level with all weights zero except for the largest tap weight. The procedure will yield different values for the same tap weights if the power distribution to the delay paths changes as described above. Another possible contribution to changes in relative weights of taps, due to changing mode patterns in the multimode fibers, is the speckle pattern of the light which emanates from the fibers. The speckle pattern is formed by the interference of light propagated in different modes in the fiber. When the mode pattern changes, the speckle pattern changes as well. Alterations in modal power distribution can change the relative response of different delay paths if the change in the speckle pattern causes a change in the total light intensity which is incident on the photosensitive areas of the photodetectors. Such an effect would be most prominent if a significant fraction of the light propagated in only a few of the lower order modes in the fiber: if a large number of modes are involved the speckle pattern appears almost homogeneous. The multimode fiber used in this demonstration probably has a few hundred modes at the 780 nm wavelength of the laser. It is very difficult to determine the relative importance of these contributions to changes in the relative responses of the delay paths since they cannot be eliminated or evaluated independently in the existing system. It is likely, however, that all such difficulties would be avoided in a system which used single mode fiber. At the short wavelengths required for the GaAs photodetectors used in this experiment the required fiber core diameter is extremely small and disadvantages would arise from low coupling efficiencies, lossy fusion splices, and the cost of a 1 X 16 single mode star coupler power splitter. This may be a productive avenue for future work using longer wavelength variable sensitivity detectors.

The transversal filter delays are formed from lengths of multimode (50/125 μm graded index) optical fiber which were connected to the outputs of the power splitter with a Northern Telecom semi-automatic fusion splicer. The fibers were cut in increments of 20 cm with an accuracy of ±0.2 cm and the outputs of the power splitter were all cleaved to the same

length ±0.1 cm prior to fusion splicing. The primary difficulty encountered in the assembly of this section of the optical distribution module was the stripping of the plastic jacket from the power splitter output fibers; this was accomplished with a mechanical fiber stripper. The power splitter had been ordered with fibers protected by a heavy duty plastic jacket and it was found to be extremely difficult, time consuming, and nerve wracking to remove the 4 cm of plastic coating required by the fiber cleaver without breaking the fiber. As a result, while the accuracy of most of the fiber delays was maintained to within ±0.5 cm, several had slightly larger errors. An additional difficulty caused by the heavy duty fiber jackets was that the automatic alignment system of the fusion splicer could not be used because its light injection module could not inject light into the heavily protected fibers. Thus, the fibers had to be aligned manually and this slightly degraded the splice quality and increased the risk that a splice with intolerable loss would created. One such splice was created and had to be redone; consequently, one of the delay paths was inaccurate by ~4 cm. Fortunately, this delay was one of the longest and since the shortest and the longest delay paths are the least important (*ie*. they have the smallest weights) for the transversal filter applications considered in this thesis, it was not deemed necessary to attempt to replace this delay path.

The sixteen fibers forming the optical delays are prepared for coupling to a photodetector array by fusion splicing them to fibers whose ends are aligned by a silicon V-groove chip. A schematic diagram of a V-groove chip with four grooves and two aligned fibers is shown in Figure 2.3. The actual V-groove chip used contains 16 grooves spaced 200 μm apart and is 5 mm long.

These V-groove fiber alignment systems are assembled by first cleaving the fibers and ensuring that the fiber jackets have been stripped back ~1 cm from the end of the fiber; this is necessary because the grooves are designed to hold bare fibers. These fibers are then gathered under a block with a slot in it wide and deep enough to hold 16 fibers side-by-side. In this way, the fibers can be separated with folded pieces of paper so that each can be pulled or pushed through the block without disturbing adjacent fibers. The holding block also serves to align the fibers coarsely so that less tension will be built up in the fibers during the fine alignment stage. The fibers are then placed in the appropriate groove one at a time and held down with the side of a hypodermic needle which is mounted to an x-y-z micropositioner; this work is performed underneath a 30X stereoscopic microscope. After each fiber is placed in a groove, the needle is advanced over it to prevent it from springing out or sliding over to an adjacent groove.

Figure 2.3. *Silicon V-groove chip with optical fibers*

After all the fibers have been placed in the grooves they are aligned with the end of the V-groove chip by gently pushing or pulling them into position; the needle holds the fibers in the grooves during this process. This alignment is important since optical crosstalk can occur if a fiber is so far from the detector array that the light emanating from it shines on more than one detector. Experience has shown that this alignment is more critical than it appears at first since light may couple from a recessed fiber to adjacent detectors through reflection paths; crosstalk attenuations as low as 20 dB in electrical power have been observed due to this effect.

Once alignment is complete a small amount of epoxy is placed on the fibers near the edge of the V-groove furthest from the fiber ends without contacting the needle; this epoxy is then cured by exposing it to high intensity ultraviolet light. The needle is then moved closer to the fiber ends and more epoxy added and cured; this process ensures that the fibers will be well seated in the grooves. Finally the needle is removed and a final coat of epoxy is applied with care being taken to ensure that epoxy does not get on the ends of the fibers. This final coat of epoxy is also applied to the part of the fiber which has had its jacket removed which extends past the V-groove chip; this strengthens the bare fiber and reduces the risk of a fiber breaking at this point in a later stage of the transversal filter assembly. With

this technique, arrays of ten fibers have been aligned to linear arrays of detectors with an overall uniformity of ±1.5 dB in electrical power. It should be noted that this figure includes the nonuniformity introduced by the optical fiber power splitter which can be as much as ±1 dB in electrical power. The V-groove assembly used in the transversal filter was one of the first to be constructed, however, and therefore it is unlikely that its measured uniformity will be as good.

2.3 The Receiver Module

The receiver module consists of a 1 X 16 photodetector array with the aligned V-groove fiber array, some circuitry to apply the dc bias voltages to the photodetectors and couple out the detected rf signal, and a wideband amplifier. The following sections will describe these components in more detail.

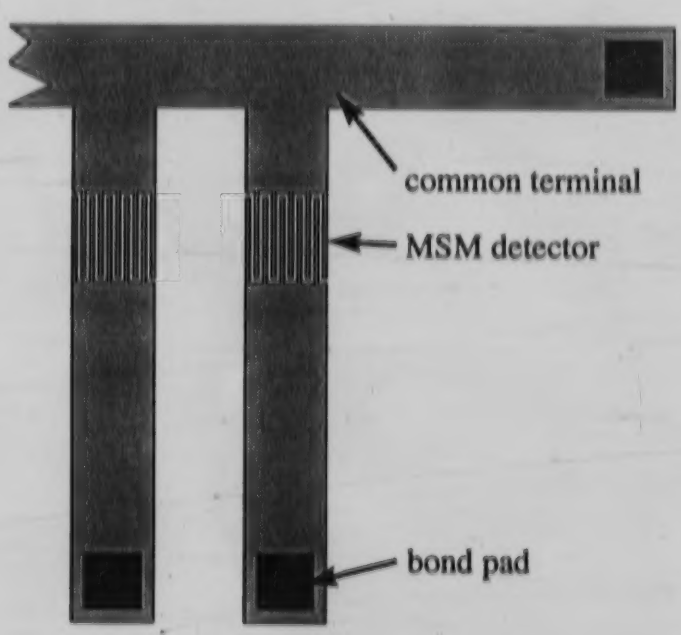

Figure 2.4. *MSM photodetector array*

2.3.1 The Photodetector Array and Fiber Alignment

The detectors used in this transversal filter demonstration are interdigital GaAs metal-semiconductor-metal (MSM) photodiodes with active areas 100 μm square, and a channel spacing of 5 μm; they are fabricated in a linear array with a center-to-center spacing of 200

μm. One contact of each detector is connected to a bond pad and the other contact is connected to a common summing bus which leads to a single bond pad. A diagram of part of the fabricated photodetector array is shown in Figure 2.4. The contacts are AuNiGe and the bond pads are gold. Experiments have shown that the responsivity of these detectors, which are from an experimental process, is about 0.1 A/W at 100 MHz. The detectors typically have a high response at low frequencies, a response which rolls off at ~8 dB/decade from ~100 MHz to ~1GHz, and a sharp roll-off at about one gigahertz. Thus, with some equalization they can be used in receivers with a bandwidth of up to 1 GHz. The V-groove fiber array is aligned with the detector array by means of a custom built fiber array clamp which is mounted on a three axis micropositioner; the fibers are mounted vertically over the detector array. A photograph of the clamp with the fibers in place is shown in Figure 2.5. The

Figure 2.5. *Fiber array clamp*

clamp consists of two metal plates connected by screws which may be used to control the pressure on the fibers. Flat pads of silicone are affixed to the metal plates to allow the clamp to grip the fibers without scratching and breaking them. The photodetector array is mounted on the receiver circuit board which in turn is mounted on a rotation stage; this provides sufficient degrees of freedom to align the fiber array accurately to the photodetector array. Al-

though the V-groove fiber array used in this demonstration had been successfully used in a previous project with adequate reported uniformity, it was noticed during the alignment that several of the fibers were apparently not well seated in their respective grooves. Thus while the coupling efficiency of each delay path could be adjusted to a fairly high level, significant loss had to be accepted to maximize the uniformity of the detector outputs. The output level of each detector with maximum bias and with all other detectors turned off is shown for one particular alignment in Figure 2.6. The spread in electrical output power of ±4.5 dB in-

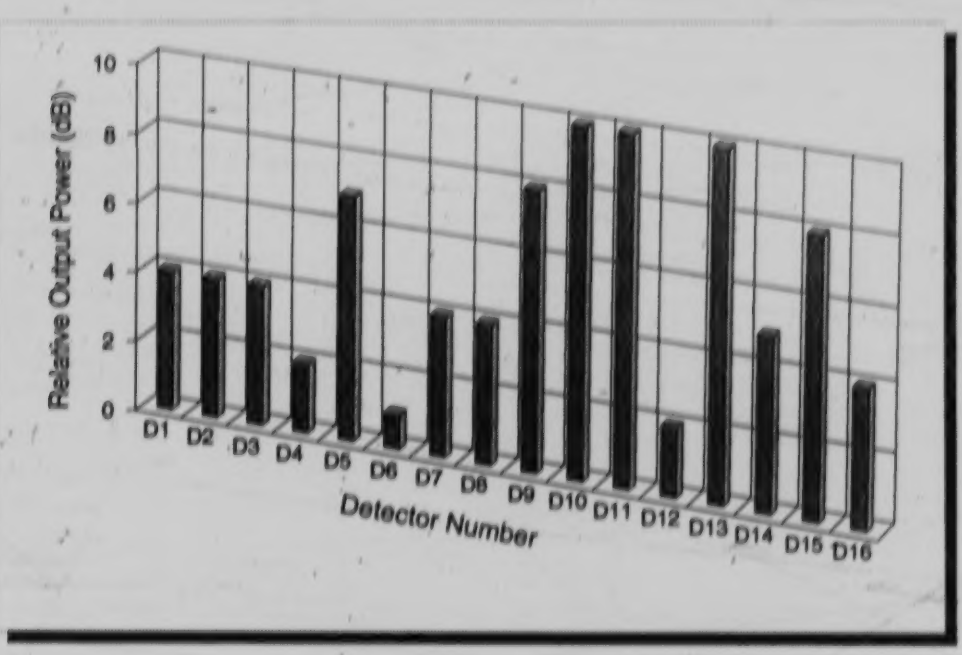

Figure 2.6. *Uniformity measurement*

cludes the uniformity of the 1 X 16 optical fiber power splitter (±1 dB electrical) and the uniformity of the fusion splices; there are two fusion splices per delay path with a combined uniformity of approximately ±1.5 dB in electrical power. The primary significance of a reduced uniformity is that it reduces the dynamic range over which the tap weights may be set. If the absolute value of the largest tap weight is set so that the response of that delay path is at the same level as the response of the weakest delay path with maximum bias then it is guaranteed that all tap weights may be implemented.

2.3.2 The Receiver Circuit

The receiver circuit is shown in Figure 2.7. The detected rf signal path is through capacitors C1 - C16 to ground and through capacitor C19 to the 50 Ω gain blocks which form the wideband post-detection amplifier.

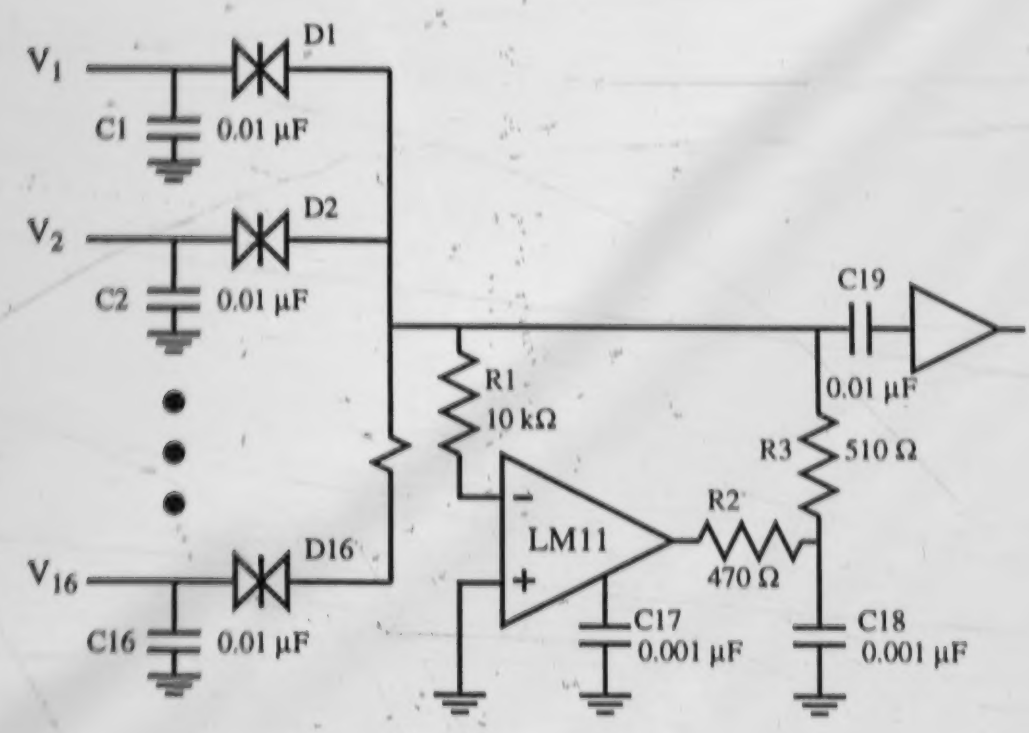

Figure 2.7. *The receiver circuit*

The MSM detectors are individually controlled by bias voltages V_1 - V_{16}. Since the bias current through an MSM detector changes when the bias voltage across it changes it is necessary to maintain the common terminal at a constant voltage regardless of the dc current passing through it. This is achieved by using the LM11 precision operational amplifier to make the common terminal a virtual ground. The LM11 maintains the common terminal at a voltage within 2 mV of ground while sourcing or sinking whatever bias current is required by the photodetectors. Capacitor C18 shunts whatever rf signal is passed by the blocking resistor R3. Compensation capacitor C17 is required for stability.

This circuit is implemented on a double-sided copper-clad printed circuit board with chip resistors and capacitors; one side of the board is used for circuit traces and the other side is used as a ground plane. Ground connections on the signal side of the board are made by drilling small holes through to the ground plane and connecting the two sides with a wire. The MSM photodetector array chip is mounted on this board with a high melting temperature wax. Connections are made to the photodetectors through ultrasonically-bonded 31.25 μm diameter germanium-doped gold wires which extend from the bond pads fabricated on the chip directly to copper pads on the circuit board. Most rf signal traces are formed from striplines with a characteristic impedance of 50 Ω which corresponds to a trace width of ~2.5 mm on the circuit boards used. The only rf signal paths which are not through 50 Ω striplines are the traces which lead from the detectors to capacitors C1 - C16.

In order to minimize the lengths of the wire bonds the sixteen copper bond pads are fabricated with a center-to-center spacing of 0.5 mm; this restricts their width to ~0.3 mm. In order to minimize the length of these high impedance traces capacitors C1 - C16 are connected to the ground plane directly by placing them side-by-side in a slot which is milled in the board; strips of mylar are used to insulate the capacitors on the signal side of the board. Solder is used to form connections between traces on either side of the capacitor and the cap of the capacitor itself; this provides a path for the dc bias current. The mask used in the fabrication of the receiver printed circuit board is shown in Appendix A.

2.3.3 The Wideband Post-Detection Amplifier

Because of the large loss associated with the optoelectronic conversion and the 1 X 16 optical power split (~60 dB in electrical power) it is necessary to amplify the detected rf signal to bring the output power up to approximately the same level as in the input power. In this transversal filter demonstration system this is achieved with a four stage wideband amplifier. Each stage consists of a commercial monolithic GaAs FET 50 Ω gain block with some associated bias and decoupling circuitry. The circuitry for the two types of gain blocks used in the amplifier are shown in Figure 2.8. The various capacitors and inductors primarily serve to decouple the rf signal from the bias path; care is taken to ensure that the decoupling is very good because it is possible for the bias path to become part of a positive feedback loop resulting in oscillations. The gain blocks have four contacts: an input, an output, and two ground contacts. Since the manufacturer suggests that it is very important that the ground contacts be of high quality they are connected directly to the ground plane through holes drilled in the circuit board. The four stages of the amplifier consist of two MSA-335 stages on the same board as the photodetector array and an INA-3170 stage fol-

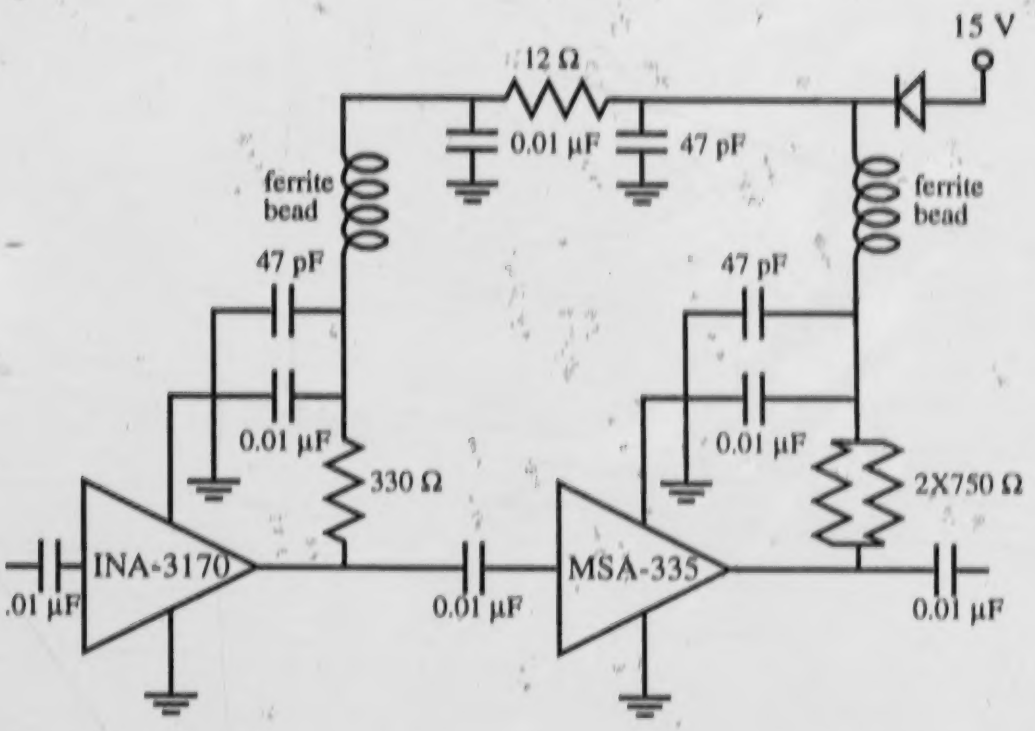

Figure 2.8. *Amplifier gain stages*

lowed by another MSA-335 stage on a separate circuit board. The two boards are mounted inside a single receiver box and are separated by a grounded brass block to prevent oscillations due to positive feedback through air coupled paths; a signal path through the brass block is provided by an SMA connector which directly contacts the 50 Ω stripline. The masks used in the fabrication of these printed circuit boards are shown in Appendix A.

The performance of the amplifier is shown in Figure 2.9. The gain of ~57 dB is consistent with the gain block's specifications of 11.5-13.5 dB for the MSA-335 and 24-28 dB for the INA-3170. The peak at 900 MHz is at least partially due to the measurement technique since it did not allow the lid of the receiver to be completely closed and this peak appeared to increase as the lid was removed. With the exception of the peak at 900 MHz, the frequency response of the amplifier is consistent with the manufacturer's specifications.

The noise figure of the amplifier can be calculated from the noise figures of the first two stages and the gain of the first stage. The specified noise figure of the MSA-335 is 6 dB.

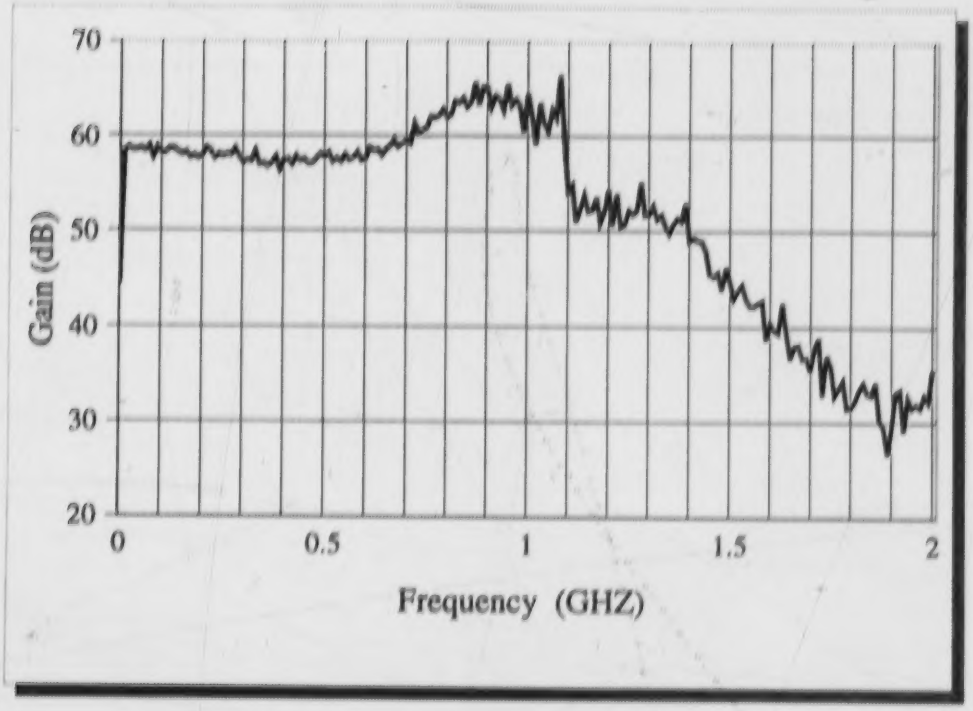

Figure 2.9. *Amplifier frequency response*

Assuming a gain of 12 dB (or 15.8), the amplifier noise figure should be 6 + 6/15.8 = 6.4 dB. The contributions to the noise figure from the third and fourth stages of the amplifier are negligible.

The noise power at the output of the transversal filter limits the performance of the filter. This power has been measured with a spectrum analyzer to be about −107 dBm in a 1 Hz bandwidth. The expected thermal noise in watts per hertz at the input to the wideband post-detection amplifier of the transversal filter receiver is $4kT$ where k is Boltzmann's constant (1.38×10^{-23} J/K) and T is the ambient temperature in Kelvin. At room temperature (290 K) this gives 1.6×10^{-20} W/Hz or −168 dBm in a 1 Hz bandwidth. Since the amplifier has a gain of ~57 dB and a noise figure of ~6 dB, and there is a loss of ~2 dB in the connectors and cables used to perform the measurement, it appears that the dominant noise source is thermal noise and can therefore the noise can be treated as white noise over the bandwidth of the filter. In the 500 MHz programmable bandwidth of the filter, there would thus be a noise power of 10^{-11} mW/Hz x 500×10^6 Hz = 0.005 mW. In a 50 Ω system this corresponds to an rms noise current of √(.000005 W / 50 Ω) = 0.32 mA,or -23 dBm, at the

output of the transversal filter. Since the maximum output power of the amplifier is 10 dBm, the maximum signal to noise ratio at the output of the filter is 33 dB. This figure could be improved by using detectors with a higher responsivity, or by using a more powerful laser, so that less gain is necessary. An improvement could also be obtained by using a front end amplifier with a lower noise figure.

Chapter 3

The Multi-Channel Analog Voltage Controller

To provide rapidly reconfigurable and precise photodetector bias voltages to set the weights of the optoelectronic transversal filter, a multi-channel analog voltage controller has been constructed. Its components include a user interface program which runs on an IBM-PC clone, a Motorola 8-bit microcontroller, and some digital-to-analog conversion, photodetector addressing, and sample-and-hold circuitry. An overview of the system is shown in Figure 3.1; the thickness of each flow line is proportional to the number of channels it represents.

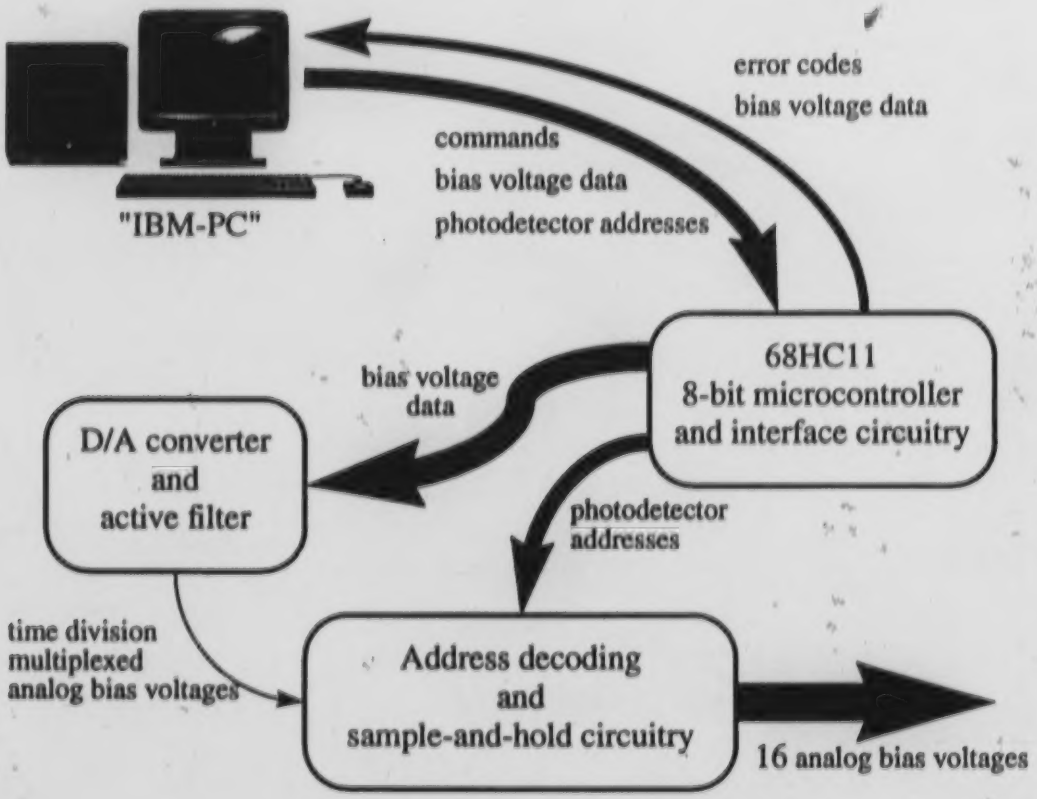

Figure 3.1. *Multi-channel analog voltage controller*

The user interface consists of a C-language program which is written in two parts: interactive code which implements a number of commands which allow the user to set one or all of the tap weights in several different ways to facilitate testing of the controller and the transversal filter, and modular code and macro definitions for communicating with the microcontroller through the computer's parallel printer port. This code may be easily incorporated into the user's own software.

The 68HC11 microcontroller performs two functions: it updates the photodetector bias voltage data stored in its memory with data provided by the user interface program, and it controls the continuous refreshing of the output analog bias voltages. Refreshing is necessary because analog sample-and-hold circuits used.

The D/A converter continuously converts the twelve bits of bias voltage data at its input to an analog voltage. While the microcontroller is in the refresh cycle this produces the bias voltages in a time division multiplexed format; this signal is demultiplexed by the address decoding and sample-and-hold circuitry to produce the sixteen bias voltages which control the transversal filter. A seven bit address and a modular output board design are used to allow the controller to produce up to 128 bias voltages. These components will be explained in more detail in the following sections.

3.1 User Interface

The software which implements several diagnostic functions and communicates user commands to the microcontroller consists of two C language files, one header file and one program file, written for execution on an IBM-PC clone (hereafter referred to as the PC). The text of these files is given in Appendix B. The means by which information is transferred between the user interface program and the microcontroller, and descriptions of each C-language file are given below.

Information is passed between the user interface and the microcontroller through an interface board containing a binary counter, data latches, flip-flops, and a number of logic gates. Commands are sent from the PC to the microcontroller by sequentially clocking data on an eight-bit data bus into a series of four octal noninverting D-type flip-flops. To signal that a command has been sent from the PC a dual J-K flip-flop with set and reset is set by the interface program to generate a non-maskable interrupt in the microcontroller. When the microcontroller has retrieved the data it resets the dual J-K flip flop. Before resetting the flip-flop the microcontroller has the opportunity to clock data from a 16 bit bus into another series of four octal flip-flops for presentation to the PC.

By monitoring the output of this flip-flop the interface program can determine when the microcontroller has finished processing the command; it may then retrieve returned data by sequentially setting the output enables of the octal flip-flops and reading the data from a four-bit data bus. The clocking of the data into the octal flip flops, the setting of the flip-flop which generates the non-maskable interrupt, and the setting of the output enables of the second set of octal flip-flops is achieved by controlling a four-bit binary asynchronous counter whose output is decoded with some logic gates to produce the appropriate signals. More information about the hardware of the interface board is given in section 3.2.1.

Communication between the interface program and the interface board is achieved through the PC's parallel printer adapter. Seventeen of the twenty-five pins of the parallel printer port correspond to specific bits in the memory of the PC; the remaining eight pins are grounded. Of these seventeen pins, eight are write-only, five are read-only, and four are read-write pins. It was decided to use the write-only pins for an eight-bit data bus for transmitting data from the PC to the microcontroller, the read-only pins for a four-bit data bus for transmitting data from the microcontroller to the PC and for directly monitoring the non-maskable interrupt pin of the microcontroller, and the read-write pins for controlling the four-bit counter. The control of the counter is achieved through direct connections to its clock pin, which advances the counter one bit, its reset pin, which sets the counter output to 0000 (binary), its load pin, which sets the counter output to the preset value which has been set to 1111, and its enable pins.

The parallel port pins may be set and read by using Turbo C's *outportb()* and *inportb()* functions, respectively. Unfortunately, this task is complicated somewhat by the fact that the bits corresponding to the pins of the parallel port are scattered across three bytes of memory in no particular order, and several of the pins are inverted with respect to the value of the bits representing them. The parallel port pin assignments are given in Table 3.1; the symbol "$" denotes a hexadecimal number, the symbol "µC" is shorthand for the microcontroller, and the bold entries denote those pins which are inverted by the printer adapter. An additional difficulty is that because of the circuitry of the printer adapter, it is possible to set and reset some of the bits faster than the pins can respond. Software has been developed which alleviates these difficulties.

The header file, *portdefn.h*, contains definitions which help make the idiosyncrasies of the IBM printer port and the details of the interface to the microcontroller transparent to the interactive user interface code. These definitions ease the development of application-spe-

Address	Label	Bit	Pin	R/W	Description
$378	DATA_OUT_PORT	0-7	2-9	W	8-bit data bus: PC to µC
$379	DATA_IN_PORT	3	15	R	µC interrupt status monitor
		4	13	R	4-bit data bus: µC to PC
		5	12	R	
		6	10	R	
		7	11	R	
$37A	CONTROL_PORT	0	1	W	Counter clock pin
		1	14	W	Counter reset pin
		2	16	W	Counter load pin
		3	17	W	Counter enable pins

Table 3.1. *Parallel port pin assignments*

cific software, such as programs which implement adaptive filter or neural network algorithms.

The first sets of definitions provide the hardware dependent addresses of the parallel port, define masks which are used to re-invert those pins inverted by the printer adapter, and define states of one of the bytes corresponding to the parallel port which provide some useful commands for the four-bit counter. The CLEAR command sets the reset pin of the counter low which causes the counter output to be set to 0000; this causes the decoding logic to set the clock line to the first octal flip-flop high which latches the data present on the eight-bit data bus. The LOAD command sets the load pin of the counter low which causes the counter output to be set to 1111 on the next rising edge on the counter's clock pin; the decoding logic responds by setting all output lines low. The RESET command sets the reset and load pins high; this is necessary to enable the counter after a CLEAR or LOAD command has been sent. This command also sets the counter clock pin low.

The next set of definitions lead to the definition of STROBE_COUNTER. This command sets the state of the counter clock pin high and then low. On the rising edge of this pulse either a load command will be executed or the counter output will be incremented by one. Because of the circuitry of the printer adapter this pin has a particularly slow rise time; if the bit corresponding to this pin is written alternately high and low on successive machine instructions on a PC with a 10 MHz clock speed the voltage of the pin will not reach the threshold required to trigger the counter. To alleviate this problem delays are introduced by

adding four *outportb()* commands; in this manner reliable clock pulses between 2 and 3 microseconds long are produced.

The final set of definitions provide commands for performing the parallel port input and output functions.

The program file, *ibm_oetf.c*, consists of several diagnostic functions and a function for writing data to the microcontroller via the eight-bit data bus; the capability of the microcontroller to return information to the user has not been implemented in the user interface software at this time. Data are sent in four eight-bit parts: a command code for the microcontroller, the address of the photodetector whose bias voltage may be updated by the data to follow, the four most significant bits of the twelve-bit bias value, and the lower eight bits of the bias value. There are several command codes recognized by the microcontroller. The UPDATE command directs the microcontroller to replace the value of the bias voltage for the indicated photodetector with the data provided. The REQUEST command directs the microcontroller to return the bias value of the photodetector indicated by the provided address. The DATA1, DATA2, and DATA3 commands direct the microcontroller to refresh the bias voltages using data from different memory blocks. In this way a single keypress of the user can cause the controller to switch between different data sets.

The function *write_data()* sends the command, address, and bias voltage data to the microcontroller interface board making use of the macro definitions contained in *portdefn.h*. By monitoring the nonmaskable interrupt line of the microcontroller the interface program can determine when the microcontroller has finished processing a command, and can avoid endless loops by detecting microcontroller malfunctions, informing the user and exiting.

The *main()* function clears the screen, prints out a menu, waits for the user to select an option, executes the option, and repeats. The available options are shown in Table 3.2. The first option allows the user to set a selected photodetector's bias voltage. The second option allows the user to increment or decrement the selected photodetector's bias voltage with single keystrokes; the amount of each increment may be specified. These two options are particularly useful for calibrating and setting the weights of the transversal filter. The third option switches the selected photodetector's bias voltage between two specified values; this is useful for determining the switching speed of the individual sample-and-hold circuits. The fourth option sets all the bias voltages to one user-specified level; this is useful in determining the output noise level and observing transients associated with the sample-and-hold circuitry. The fifth option sets all the bias voltages; the user specificies values for the

Option Number	Description
1	Set a specific crosspoint's bias
2	Increment/Decrement bias
3	Alternate bias between two values
4	Set all crosspoint biases to one value
5	Set all biases alternately to two values
6	Uniformly space all biases between two values
7	Select data set 1
8	Select data set 2
9	Select data set 3

Table 3.2. *User interface options*

even and odd numbered outputs separately. This command is useful for testing the controller's worst-case rise time and overshoot. The sixth option sets all of the bias voltages so that they are uniformly spaced between two user-specified values; this creates a stepped voltage ramp on the analog output of the D/A converter. With this option the feed-through isolation of the sample-and-hold circuits may be conveniently measured. The final three options allow the user to switch between three independent sets of bias voltage values stored in the microcontrollers memory. This option facilitates the calibration of the transversal filter and allows the switching speed of the transversal filter to be demonstrated.

3.2 68HC11 Microcontroller and Interface Circuitry

The digital section of the multi-channel analog voltage controller is based on an eight-bit microcontroller, the 68HC11. This section consists of the 68HC11 evaluation board, which emulates a 68HC11 and has the capability of running assembly language programs downloaded from an external computer, and two wire-wrapped interface boards: one for communicating with the user interface program, and one primarily for controlling the output of bias voltage and photodetector address data. The design and construction of the interface boards, and the programming of the 68HC11 were largely completed by an undergraduate student, Shreyas Paranjpe, and the details of his work are given in [32]. As a number of bugs have been uncovered and corrected, and some design changes implemented since his work was completed, an updated description of the work is presented here.

3.2.1 IBM-PC to 68HC11 Interface Board

The IBM-PC to 68HC11 interface board handles the transfer of data between the user interface program and the 68HC11 assembly language software. A block diagram of the board is shown in Figure 3.2; a detailed schematic diagram can be found in Appendix C.

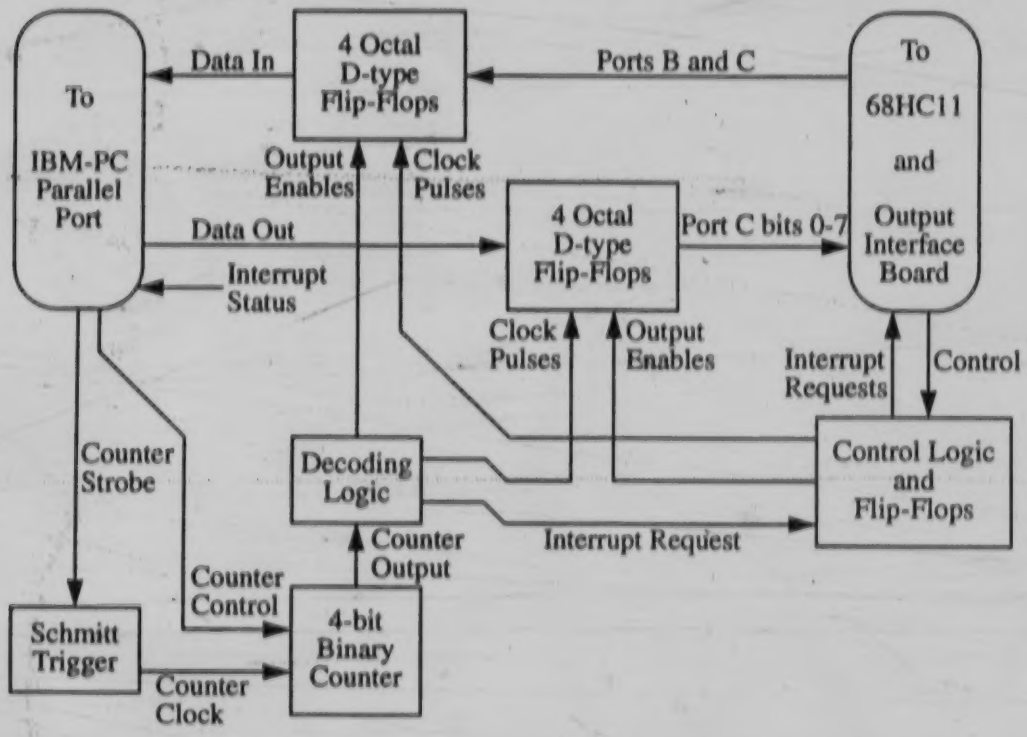

Figure 3.2. *IBM-PC to 68HC11 interface board*

The components used to implement the design are high speed CMOS integrated circuits; this type was chosen for its low power consumption and high immunity to electromagnetic interference (EMI). Unfortunately, since this technology involves the generation of very short current spikes it is also characterized by high levels of emitted EMI, necessitating the shielding of other components of the controller. Information is passed between the 68HC11 and the PC through octal D-type flip-flops; four are used for each direction of transmission. The transmitting system provides the appropriate clock pulses to latch data on to the flip-flops and the receiving system provides the output enables to the flip-flops to retrieve the data. On the PC side the control signals are generated by a 4-bit binary counter and some decoding logic gates. Rather than using the pins of the parallel port directly, a counter is

used to provide the input to the decoding logic to avoid timing problems and the resulting generation of false signals. On the 68HC11 side the control signals are generated by some logic gates and flip-flops; the inputs to the logic gates are generated by the 68HC11's STRB pin and a 4 X 16 decoder which resides on the other interface board (described in section 3.2.2) where it is directly controlled by four pins of the 68HC11's port D. Data are retrieved through the microcontroller's port C and data are transmitted through its ports B and C; port B is an eight-bit output port, port C is an eight-bit input/output port, and port D is a six-bit output port.

3.2.2 Output Interface Board

The output interface board controls the output of photodetector address and bias voltage data; it also generates some of the control signals for the IBM-PC to 68HC11 interface board. A block diagram of the board is shown in Figure 3.3; a detailed schematic diagram can be found in Appendix C.

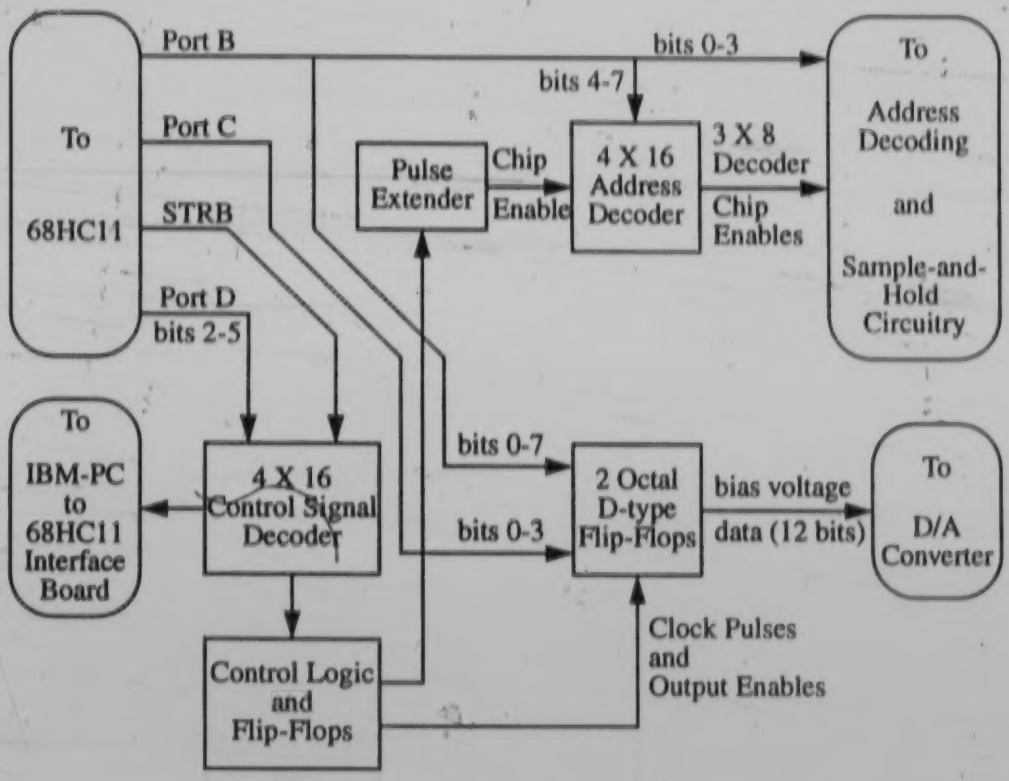

Figure 3.3. *Output interface board*

The refresh cycle begins with the address decoding circuitry disabled; as a result all the sample-and-hold circuits will be in the "hold" mode. Twelve-bit bias voltage data are then written to the two octal D-type flip-flops through the microcontroller's port B and port C. The least significant bit is port B's bit 0 and the most significant bit is port C's bit 3. The data are latched on to the flip-flops by sending the appropriate four-bit command to the 4 X 16 control signal decoder and causing the microcontroller's STRB pin to output a pulse; this causes the control logic to generate the required clock pulse for the D-type flip-flops. The flip-flops are continuously output enabled so that conversion of the data begins as soon as possible.

Port B is then used to deliver the seven-bit photodetector address to the address decoding circuitry. The four most significant bits are sent to a 4 X 16 decoder which produces chip enables for a series of 3 X 8 decoders to which the other three bits of the address are sent. The address decoding circuitry may be disabled by disabling the 4 X 16 decoder which is located on the output interface board. Since each sample-and-hold circuit only samples while this decoder is enabled, and the decoder chip enable is controlled by the STRB pin on the 68HC11 which produces pulses 1 μs in duration, a pulse extender has been added since the original work was completed to ensure that decoder chip is enabled for as much of the refresh cycle as possible. It is formed from a 74HC4538 precision monostable multivibrator in a nonretriggerable configuration. Its time constant has been set so that a falling edge on the input signal causes it to output a low pulse for ~40 μs, enabling the address decoding circuitry and causing one of the sample-and-hold circuits to sample the D/A converter's output; the amount of time each bias voltage is present at the output of the D/A converter is currently set at ~45 μs.

3.2.3 68HC11 Microcontroller Software

The tasks of refreshing the output voltages of the sample-and-hold circuits and responding to user commands are performed by the 68HC11 microcontroller software. The software consists of a series of assembly language program files, listed in Table 3.3, which are compiled and linked on an IBM-PC clone using a cross-assembler provided by Motorola. The text of these files is given in Appendix B. The cross-assembler produces an S-record format machine language file which must then be downloaded to the 68HC11 evaluation board each time the system is powered up; detailed instructions for compiling the software, downloading the S-record file to the 68HC11, and beginning execution of the program are given in Appendix D. After initialization the program enters an endless loop in which it sends instructions to refresh the output voltages from address 99 to 0. When an

Filename	Program Function
initial.doc	Initializes internal registers and external flip-flops
declare.doc	Declares constants, variables, and data arrays
refresh.doc	Refreshes output voltages
intseq.doc	Handles user interface commands
straint.doc	Retrieves user interface commands
updseq.doc	Performs user interface UPDATE command
reqseq.doc	Performs user interface REQUEST command
loaddata.doc	Unused at present
error.doc	Returns error if user command unrecognized

Table 3.3. *68HC11 assembly language files*

interrupt request is received from the user interface program, the interrupt routine contained in the file *intseq.doc* is executed. This routine retrieves the transmitted data by calling the routine contained in *straint.doc* and executes the specified command. Further details regarding this software can be found in [32].

Two modifications have been made during the evolution of the design: delays have been introduced to lengthen the time of the refresh cycle from 20 μs to 50 μs in order to allow the output circuits to make longer, more accurate samples of the D/A converter's output; also, a serious bug was traced and corrected. The problem was that the state of the flip-flop which controls the clocking of bias voltage data into the octal D-type flip-flops is implicitly toggled during the refresh cycle but the interrupt routine explicitly sets the flip-flop's state low. Since the flip-flop's state was high for three of eleven instructions that made up the refresh cycle, approximately twenty percent of all the non-maskable interrupt requests resulted in incorrect data being latched and sent to the D/A converter. The problem was solved by modifying the interrupt routine so that it pulls the value of the program counter off the stack and checks to see if the interrupt occured when the state of the flip-flop was high; if so, it explicitly sets the flip-flop's state high just before returning from the interrupt. With this modification, all components of the digital section of the controller were found to perform as expected.

3.3 Digital-to-Analog Converter

The purpose of the digital-to-analog converter board is to provide an accurate conversion of the twelve-bit bias voltage data to an analog signal between +5 and -5 volts in no more than four microseconds and with as low a noise level as possible. This is achieved

with a high speed twelve-bit monolithic D/A converter, a transimpedance amplifier to produce a buffered voltage output, and a two-stage low-pass active filter to reduce noise; a schematic diagram is given in Appendix C. The circuit is implemented on a double-sided printed circuit board mounted inside a grounded die-cast aluminium box which is itself mounted on a grounded copper-clad board sized to fit the controller card cage. The box serves to isolate the D/A converter circuitry from electromagnetic interference and the card cage mounted board is used to help isolate the controller's analog circuitry from the digital circuitry associated with the microcontroller. To minimize EMI the circuit board is designed with power/ground loops of minimal area (to reduce the inductance of the current loops) and the output of the board is delivered to the sample-and-hold circuitry through a coaxial cable. The masks used to manufacture the printed circuit board, and component layout diagrams are given in Appendix A.

The voltage data are delivered from the output interface board to the D/A converter board through a twenty-six conductor ribbon cable; alternate conductors in the ribbon cable are grounded to minimize the area of the current loops and thus the effects of inductive pick-up. Before being delivered to the D/A converter, the data lines are each passed through a low pass circuit consisting of a 100 Ω series resistor and a 56 pF capacitor to ground. The circuit serves to dampen overshoot and ringing effects and filter high frequency noise. The resistor value was chosen so as to not overload the output current source specification of the octal D-type flip-flop and the capacitor value was chosen empirically to provide a fast rise time with minimal overshoot.

Since the D/A converter provides a current output, a high slew rate op-amp in a transimpedance configuration is necessary to produce a voltage output; the D/A converter package provides the feedback resistor and the voltage reference control necessary to produce a bipolar output with a ten volt span. The compensation capacitor of the op-amp has been empirically adjusted to produce the minimum rise time in response to a ten volt step with negligible overshoot; this configuration has a measured maximum slew rate of 5 V/μs.

Tests performed with a prototype circuit board indicated the need for a low pass filter to reduce further the effects of EMI from switching transients generated by the digital section of the controller. This has been accomplished by cascading two standard Sallen and Key noninverting low pass active filter stages; one such filter stage is shown in Figure 3.4. In the case where $R_1 = R_2 = R$ and $C_1 = C_2 = C$ the transfer function of this circuit can be expressed as

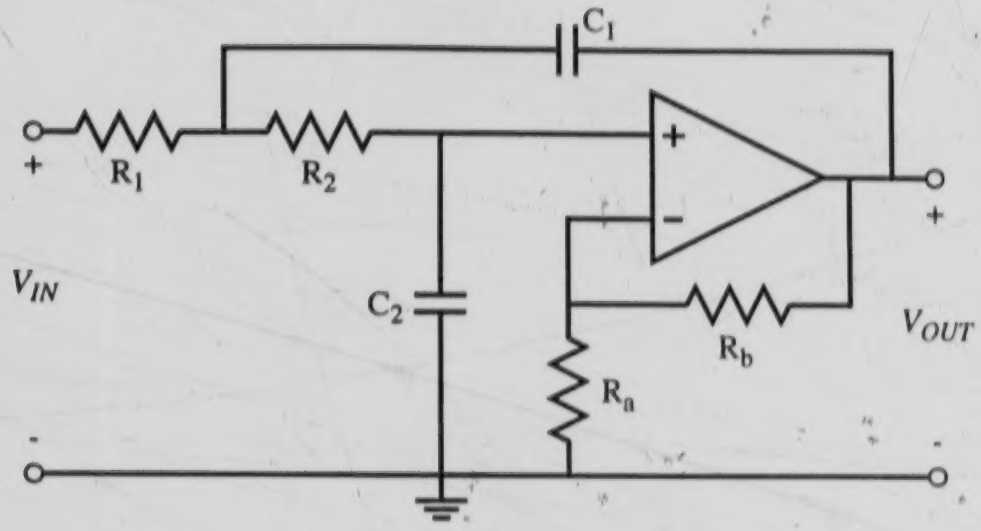

Figure 3.4. *Sallen and Key low-pass noninverting amplifier section*

$$H(s) = \frac{A_v}{R^2C^2s^2 + RCs(3 - A_v) + 1} \tag{3.2}$$

where the gain A_v is given by

$$A_v = 1 + \frac{R_b}{R_a} \tag{3.3}$$

and the 3 dB cut-off frequency, w_o, is given by

$$w_o = \frac{1}{RC}. \tag{3.4}$$

In this case the filter was chosen to have unity gain so resistor R_a is replaced by an open circuit and resisitor R_b is replaced by a short circuit. Since it was found that most of the interfering noise appeared at frequencies in the low tens of megahertz, the filter was designed with a 3 dB cut-off frequency of 1 MHz for each stage. Choosing the value of the resistance to be 2210 Ω and the value of the capacitance to be 68 pF satisfies equation (3.3). The op-amp used is the TL072 which is a dual low-noise JFET input op-amp with a slew rate of 13 V/μs and a unity gain bandwidth of 4 MHz. Although this circuit's performance will not be ideal above 4 MHz, a SPICE simulation and the circuit's observed performance

seem to indicate that it gives attenuations of at least 25 dB in the low tens of megahertz. The delay introduced by this circuit is such that a ten volt step at a slew rate of 5 V/μs produces an output voltage within 0.1% of its final value after 4 μs; this was predicted by simulation and confirmed by experiment. The completed D/A converter board, as shown by experiment, is capable of converting twelve bit data to an analog voltage between -5 and +5 volts with an accuracy of at least 0.1% in 4 μs with a peak-to-peak noise level of less than 5 mV as observed with a 350 MHz bandwidth oscilloscope.

3.4 Address Decoding and Sample-and-Hold Circuitry

The purpose of the address decoding and sample-and-hold circuitry is to provide stable, low-noise, rapidly reconfigurable output channels. This is achieved by using separate sample-and-hold circuits for each output channel. Schematic diagrams are given in Appendix C. The voltage maintained by each sample-and-hold circuit is refreshed periodically by causing the circuit to sample the output of the D/A converter board at the appropriate time. The sample control signals are generated by 3 X 8 decoders whose output is selected by the lower three bits of the address delivered from the output interface board; the chip enables of these decoders are themselves controlled by the output of the 4 X 16 address decoder on the output interface board.

The sample-and-hold sections consists of LF398 monolithic sample-and-hold circuits, each with a hold capacitor and a potentiometer for adjusting the output offset voltage. The value of the hold capacitor is determined primarily by a compromise between sampling accuracy and output droop rate; the value of 0.022 μF chosen gives a sampling accuracy of ~0.1% in the ~40 μs sample time per refresh cycle, and a droop rate of ~5 mV per second. The worst case sampling offset is 10 mV and worst case droop is negligible given a refresh period of 5 ms (50 μs per output x 100 outputs). The peak-to-peak output noise level has been measured as less than 5 mV in a 350 MHz bandwidth; this level is near the instrument floor of the oscilloscope and may actually be lower. The LF398 is capable of sourcing or sinking up to 4 mA which is sufficient bias current for the photodetectors used in the transversal filter (under illumination with a 5 V bias they typically draw 100 μA).

An unfortunate characteristic of the LF398 which had to be dealt with is the presence of a transient spike several hundred nanoseconds long in the output voltage when the circuit switches from the hold mode to the sample mode. The effect of this transient is reduced by placing a 1 kΩ resistor in series with each output. In conjunction with the 10 nF capacitor to ground in parallel with each photodetector this forms a low pass filter which spreads the

energy of the transient over several tens of microseconds; SPICE simulation and experimental results indicate that this approach reduces the maximum value of the transient to ~10 mV. Since the bias current drawn by the photodetectors used in the transversal filter depends in a nonlinear way on the bias voltage there will be a bias voltage dependent voltage drop across the resistor in series with the sample-and-hold circuit output which will reduce the linearity of the output voltage as a function of the data. Fortunately this is not a concern in the transversal filter since the weight which is being applied depends in a nonlinear way on the bias voltage to begin with, and with effort can be taken into account in the calibration of the detectors.

The address decoding and sample-and-hold circuitry is implemented on a printed circuit board to facilitate construction and maintenance; masks used in the board's fabrication and a diagram of the layout of components on the board are given in Appendix A. To increase the isolation between the digital and analog sections of the board, digital and analog grounds are kept separate and power/ground loops are designed to have minimal area. Twenty-four sample and hold circuits, and thus three 3 X 8 decoders, are contained on each board; four boards have been constructed to date to produce 96 output channels. Digital inputs are through fourteen conductor twisted pair ribbon cable with one conductor of each pair connected to ground. The analog input is made through an edge-mounted BNC connector to an RG/58 50 Ω coaxial cable. The analog outputs are delivered through fifty conductor twisted pair ribbon cable with one conductor of each pair connected to ground to minimize current loop area.

In its present state, the analog voltage controller can independently set up to 96 voltages over a range of ±4.9 volts with at least 0.2% accuracy. All 96 voltages can be reconfigured as a set in 5 ms (50μs per output channel); the fifteen tap weights of the transversal filter can thus be reconfigured in 750 μs.

Chapter 4

System Performance

This chapter describes the performance of the combined analog voltage controller and transversal filter system. The ability of the system to produce variable bipolar weights is investigated, and the results of experiments to synthesize bandpass filters with fifteen of the taps of the transversal filter are reported.

4.1 Optoelectronically Set Weights

4.1.1 Variable Bipolar Weights

By adjusting the dc bias voltage across an MSM photodetector we control its responsivity. The responsivity as a function of bias voltage of a particular detector is isolated by first adjusting the bias voltages of all sixteen detectors in the array to minimize their combined response to a 100 MHz sine wave generated by an HP8753A network analyzer. Then the bias voltage of one of the detectors is swept from -5 to +5 volts and the transversal filter output as measured with the network analyzeris recorded; the results of such an experiment are shown in Figure 4.1. Note that the recorded output level has been amplified and current values have been calculated from the collected data assuming a 50 Ω load impedance. Negative values of the root-mean-squared output current indicate a phase shift of 180° relative to the positive values. The average optical power incident on the photodetector used in this experiment is ~0.2 mW and the modulation index is nearly one. It should be noted that when the magnitude of the bias voltage is small, say less than 0.1 V, the curve of Figure 4.1 becomes less representative of the sensitivity of the photodetector because the noise level and the residual responses of the other detectors begin to dominate the output of the transversal filter. Experiments have shown that the responses of detectors in the same array have a very similar dependence on the bias voltage; in particular the value of the bias voltage required to minimize the responsivity of each detector is the same to within 1 mV.

The most significant feature of this Figure 4.1 is that it indicates that a 180° phase shift in the output can be obtained by simply reversing the polarity of the bias. This allows the implementation of bipolar weights with a single delay path for each weight value. Previous demonstrations of optical fiber transversal filters with bipolar weights required the use of either expensive coherent technology, or, if the weights were applied by variable attenua-

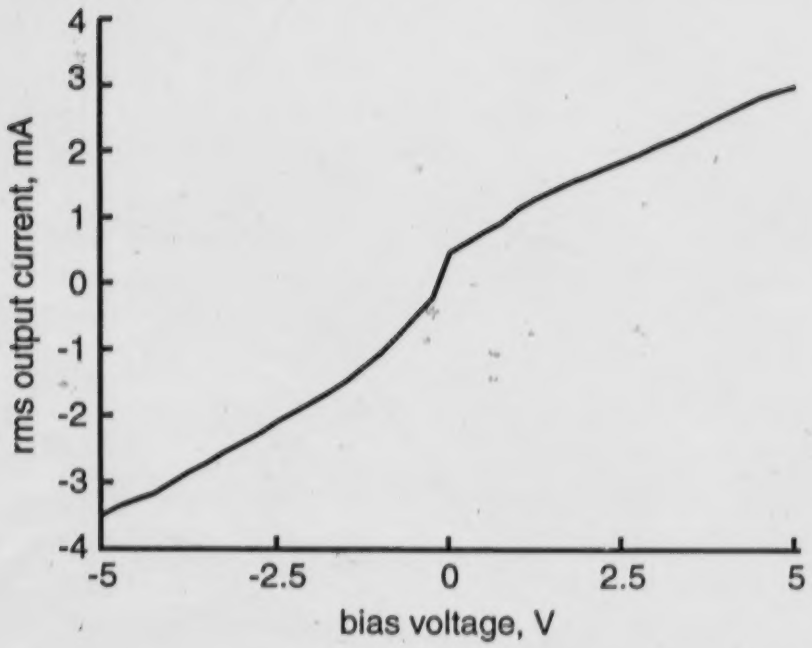

Figure 4.1. *Transversal filter output as a function of one bias voltage*

tors, two delay paths for each weight value. The outputs of the "positive" and "negative" optical delay paths could then be subtracted by using two detectors placed back to back. Such schemes introduce a 3 dB loss in optical power in each delay path and often add significant cost and bulk to the system.

4.1.2 Weight Accuracy

An important parameter characterizing the performance of transversal filters is the accuracy with which the weights can be set. A number of factors which influence this parameter in the system which has been constructed will be described in the following paragraphs.

One possible limitation on the weight accuracy is the precision with which the bias voltages can be set. As was indicated in Chapter 3, the controller has been designed to provide voltages over a ten volt range with a worst case precision of 0.2%, or 20 mV. The weight precison that this corresponds to depends on the slope of the curve in Figure 4.1. Over most of the voltage range the slope is ~0.6 mA/V so a 20 mV imprecision in the bias voltage corresponds to an imprecision of 0.012 mA in the rms output current; given the ~3 mA unipolar range of the rms output current this gives a weight imprecision of 0.4%.

Note, however that as the bias voltage approaches zero the slope of the curve in Figure 4.1 increases by a factor of three or four and the weight imprecision increases likewise.

An important consequence of the imprecision in the weight values is that the photodetectors cannot be completely turned off. This implies that attempts to set weight values that are small (less than ~5% of the maximum rms current output) using data of the type presented in Figure 4.1 could be even more imprecise than the ~1.6% indicated by the slope of the curve since the assumption that these data represent the responsivity of a single detector is no longer valid: at low bias levels the transversal filter output is dominated by the residual responses of the other fifteen detectors.

A possible contributor to the degradation in the accuracy with which the weights can be set is optical crosstalk at the detector array. Tests with other V-groove assemblies and identical detector arrays have shown that light from one fiber may be coupled into the detectors adjacent to the target detector with attenuations as small as 30 dB in electrical power. The precise level of optical crosstalk of the assembled transversal filter was impossible to measure, since the modulated optical signal could not be selectively turned on and off for individual detectors. If the crosstalk of the assembled system is similiar to that of the other systems tested using similar detector arrays [33], however, the contribution to the weight inaccuracy should be less than 3%.

The accuracy with which the weights can be set will be degraded if the shape of the frequency response of the detector is not independent of bias voltage. The frequency response of the MSM photodetectors used in this experiment at four different bias voltages is shown in Figure 4.2. It is clear that the frequency response of the detectors drops off more steeply for the smaller bias voltages. At 100 MHz, the difference between the response with a 5 V bias and a 0.5 V bias is 15 dB, while at 500 MHz the difference is 17.5 dB. If the largest bias voltage is chosen to be the reference voltage which is used to calibrate the transversal filter weights, then the smallest weights will be most affected by the error introduced by this characteristic of the detectors. Also, the degradation in the accuracy with which the weights are set will be higher for frequencies further from the frequency which is used to calibrate the weights. This effect degrades the accuracy by up to ~5% over the 500 MHz programmable bandwidth of the filter.

The time variation of the power divison ratio of the 1 X 16 optical fiber power splitter that results from mode hops may also contribute to the tap weight inaccuracy. While annoying, this effect is relatively minor and can be eliminated by adjusting the bias current of la-

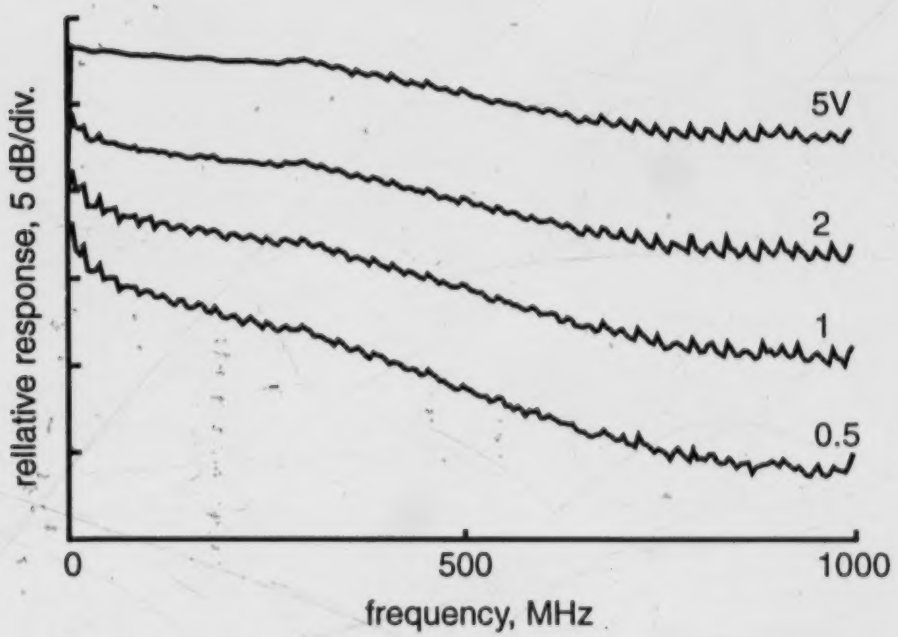

Figure 4.2. *MSM photodetector frequency response for various bias voltages*

ser as required to force it to return to the initial output mode structure, and thus the power division ratio, which is being used to calibrate the weights.

Since there are bias level, bandwidth, frequency, and time dependent factors which affect the accuracy with which the transversal filter tap weights can be set, this accuracy is difficult to quantify. Adding the various sources of degradation yields a worst case accuracy of 10%; the typical accuracy which can be expected is ~5%. The principal sources of degradation are the dependence of the shape of the detector frequency response on bias voltage, and optical crosstalk at the detector array.

4.2 Bandpass Filter Synthesis

To demonstrate the optoelectronic transversal filter, it was used to synthesize linear phase bandpass filters. This section describes the means by which the tap weights were set, the results of the synthesis experiments, and a demonstration of the agility of the system.

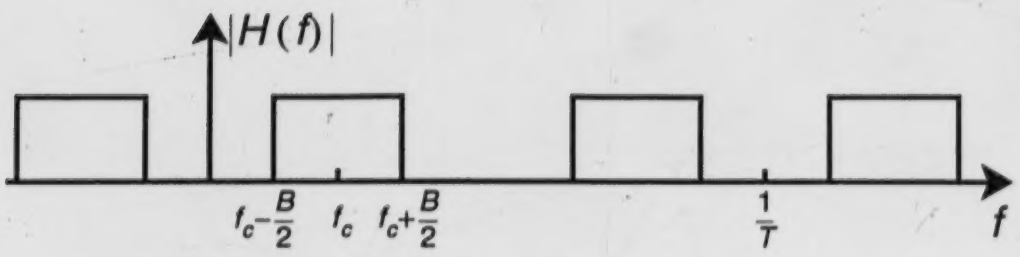

Figure 4.3. *Ideal periodic bandpass filter function*

4.2.1 Tap Weight Determination

The ideal magnitude response of the periodic function being synthesized is shown in Figure 4.3; f_c is the center frequency of the passband, B is the bandwidth of each passband, and $1/T$ is the period in the frequency domain (T is the incremental tap delay). Note that this is an even function in the frequency domain. The phase response of the ideal function is to be some linear function of frequency. As was indicated in Chapter 1, this frequency response can be approximated by

$$H(f) = [e^{-j(N+1)(\pi T) f}] \left[w_o + 2 \sum_{m=1}^{M} w_m \cos(2\pi m T f) \right] \tag{4.5}$$

where in the case of a 15-tap synthesis $N = 15$, $M = 7$, and the Fourier coefficients w_m are given by

$$w_m = 2T \int_{f_1}^{(f_1 + 1/T)} H(f) \cos(2\pi m T f)\, df \tag{4.6}$$

for $m = 0, 1 \ldots 7$. Since $H(f)$ is an even function, $w_m = w_{-m}$. Evaluating equation (4.2) for the function shown in Figure 4.3 yields

$$w_m = \frac{2}{m\pi} \left\{ \sin\left[2\pi m T \left(f_c + \frac{B}{2}\right)\right] - \sin\left[2\pi m T \left(f_c - \frac{B}{2}\right)\right] \right\}. \tag{4.7}$$

Note that since the series is truncated, equation (4.3) gives the weights for a rectangular window of the Fourier coefficients; with such a window, the ripples of the Gibbs phenomenon near the discontinuities in $H(f)$ give rise to high sidelobes in the synthesized filter. For this reason, triangular, raised cosine, and Gaussian window functions were also exam-

	f_c = 200 MHz		f_c = 350 MHz	
m	w_m	$w_m e^{-\alpha m^2}$	w_m	$w_m e^{-\alpha m^2}$
0	1	1	1	1
1	0.308	0.305	-0.585	-0.581
2	-0.796	-0.771	-0.304	-0.294
3	-0.779	-0.725	0.916	0.853
4	0.289	0.254	-0.757	-0.666
5	0.900	0.737	<.001	<.001
6	0.265	0.199	0.694	0.521
7	-0.656	-0.443	-0.771	-0.521

Table 4.1. *Bandpass filter Fourier coefficients with Gaussian window*

ined. Of these window functions it was found that the greatest sidelobe reduction was obtained from a Gaussian window formed by multiplying the Fourier coefficients by $e^{-\alpha m^2}$ where the parameter $\alpha = 0.008$. The search of window functions was not exhaustive and was only intended to find a simple function that would provide significant additional sidelobe attenuation.

Using equation (4.3) and the Gaussian window function, the Fourier coefficients were calculated for the cases with B = 50 MHz, T = 1 ns, and f_c = 200 and 350 MHz; the normalized results are shown in Table 4.1. The calculated power spectrum corresponding to the synthesis of the bandpass filter with a center frequency of 200 MHz using the rectangular window is shown in figure 4.4. Close examination of this figure reveals that the highest sidelobe level is 12.2 dB down from the main peak. The calculated spectrum of the synthesis of the bandpass filter with a center frequency of 200 MHz using the Gaussian window is shown in Figure 4.5. The highest sidelobe level is 15.6 dB down from the main peak.

4.2.2 Relative Importance of Tap Weight and Tap Delay Accuracy

In this section, the relative importance of the imprecison in the tap weights and the errors in the delay lengths will be examined qualitatively by showing the typical effects of randomly distributed errors in the tap weights and delay lengths..

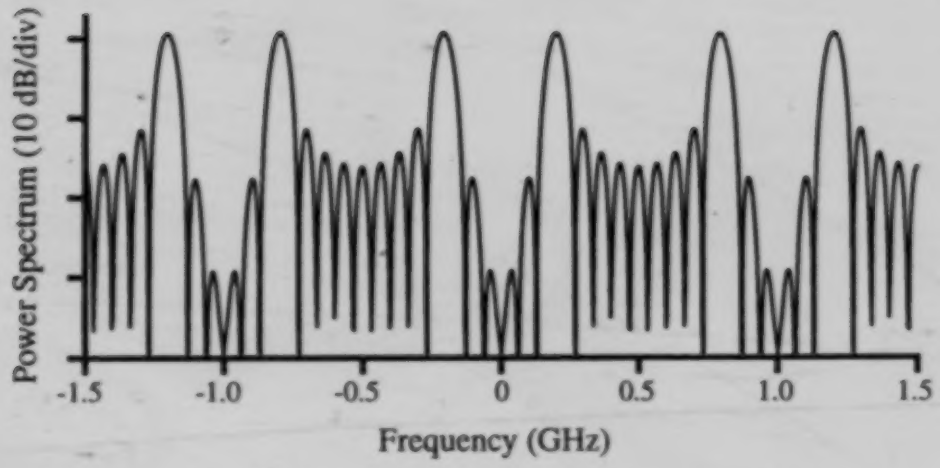

Figure 4.4. *Calculated power spectrum of synthesized bandpass filter centered at 200 MHz using a rectangular window function*

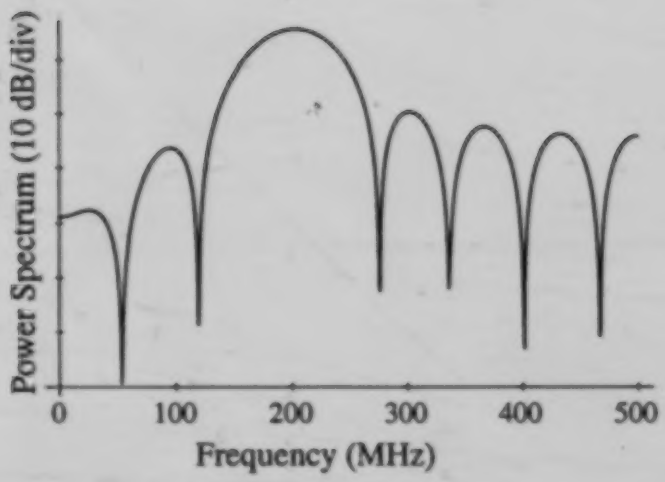

Figure 4.5. *Calculated power spectrum of synthesized bandpass filter centered at 200 MHz using a Gaussian window function*

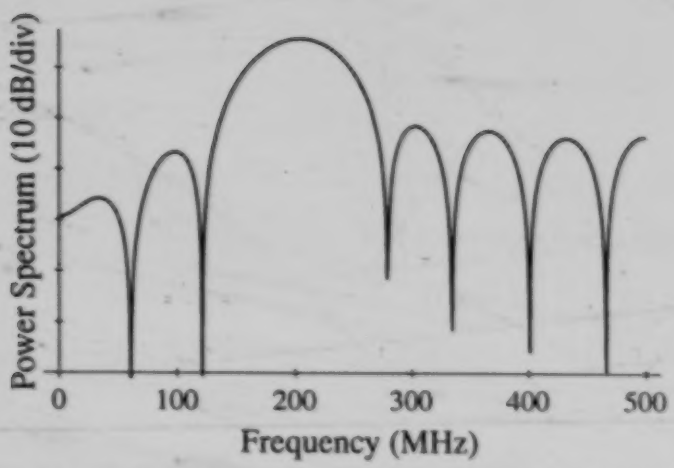

Figure 4.6. *Calculated power spectrum of synthesized bandpass filter centered at 200 MHz with up to a 10% error in the weights*

Figure 4.6 shows the typical effect of imprecisions in the weight magnitudes on the synthesis of the bandpass filter centered at 200 MHz, whose Gaussian-windowed Fourier coefficients are given in Table 4.1. The weight errors are randomly distributed between 0 and 10%. The same synthesis with up to a 25% error in the weights is shown in Figure 4.7. Clearly, the attenuation of the first sidelobe is relatively insensitive to errors in the weights, at least for this particular example. The primary effect of weight imprecision in the synthe-

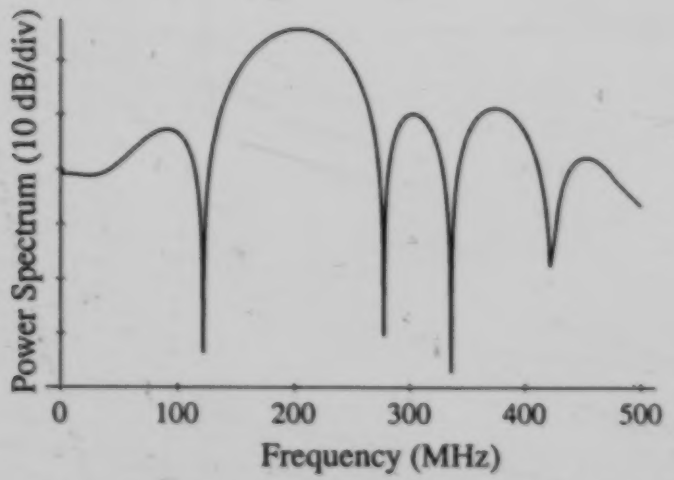

Figure 4.7. *Calculated power spectrum of synthesized bandpass filter centered at 200 MHz with up to a 25% error in the weights*

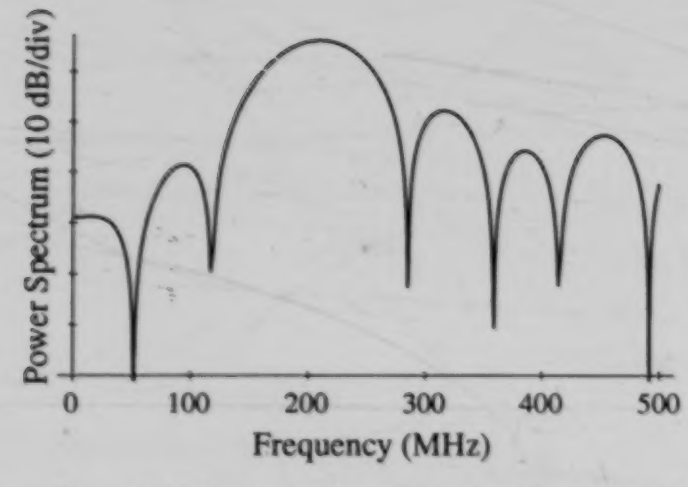

Figure 4.8. *Calculated power spectrum of synthesized bandpass filter centered at 200 MHz with up to a 10% error in the tap delays*

sis of the bandpass filters demonstrated in this thesis appears to be in changing the sidelobe structure.

The typical effect of errors in the delay lengths is shown in Figure 4.8 for the same bandpass filter centered at 200 MHz. The errors are randomly distributed between 0 and 10% of the incremental tap delay. Figure 4.9 shows the typical effect of errors of up to 25%

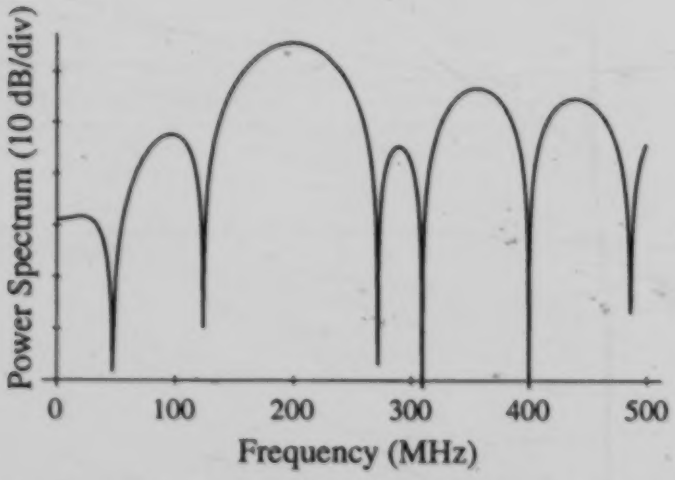

Figure 4.9. *Calculated power spectrum of synthesized bandpass filter centered at 200 MHz with up to a 25% error in the tap delays*

of the incremental tap delay. Comparison with Figures 4.6 and 4.7 shows that the filter response is typically more sensitive to errors in the tap delays.

In the transversal filter which has been constructed the typical inaccuracy in the magnitudes of the weights is ~5% over the 500 MHz bandwidth of the filter. The typical error in the tap lengths is estimated to be less than ±0.5 cm, or 2.5% of the 20 cm incremental tap length. With these parameters the above simulations suggest that there will be negligible degradation of the frequency response in the passband, and on the order of 2 or 3 dB reduction of the main peak to sidelobe attenuation ratio. There are, however, two or three errors of 5 to 10% and one of ~25% of the incremental tap length. Depending on the value of the weights of these delay paths, significant degradation of the synthesized response may be observed.

4.2.3 Experimental Results

Using the data of Table 4.1, 50 MHz bandpass filters at center frequencies of 200 MHz and 350 MHz were synthesized with the optoelectronic transversal filter. The procedure by which the multi-channel analog voltage controller was used to set the transversal filter weights is as follows. First, the response of each delay path to a sine wave at the center frequency of the bandpass filter was measured to see which path gave the weakest response under maximum bias; typical data collected from such an experiment is shown in Figure 2.6. This weakest response level was chosen to be the reference level. Then, with all other bias voltages set to zero, the voltage corresponding to the center tap weight was adjusted so that the output was at the reference level. The controller was then instructed to refresh the bias voltages with a different data set. The individual tap weights were set by adjusting the corresponding bias voltages one by one, with all other bias voltages set to zero, and comparing the output to the reference level. This comparison could be achieved at the push of a button by instructing the controller to switch data sets. In this manner, the effect of the time variation of the output could be minimized. Since three data sets are available, it was possible to use this procedure to set and store the weights for both bandpass filters simultaneously.

The results of the bandpass filter synthesis are shown in Figure 4.10. The data are collected with an HP8753A Network Analyzer with the IF bandwidth set to 300 Hz. The data are not corrected for the frequency responses of the laser, detectors, and amplifier; the combined frequency response of these devices is shown in Figure 4.2. The first sidelobe level for the bandpass filter centered at 200 MHz is 15.5 dB below the main peak; for the bandpass filter centered at 350 MHz, it is 12.4 dB. It should be noted, however, that the slight

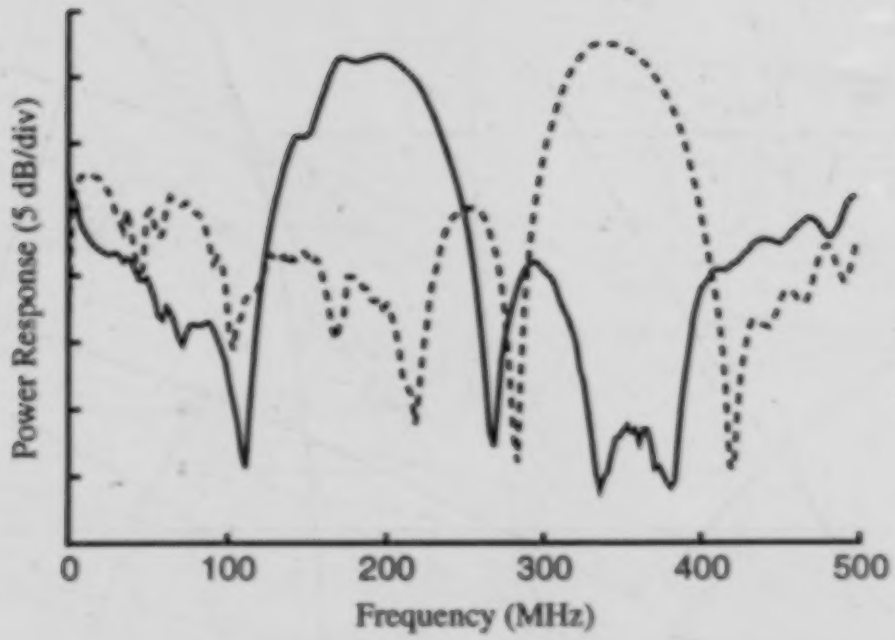

Figure 4.10. *Measured power responses of synthesized bandpass filters centered at 200 MHz (solid line) and 350 MHz (dashed line)*

roll-off shown in Figure 4.2 will serve to increase the observed minimum first sidelobe attenuation in the former case, and will serve to decrease the attenuation in the latter case because of the position of the highest first sidelobes relative to the main peak. Thus, the attenuation due solely to the transversal filter response is between 13 and 14 dB which is quite close to the calculated value of 15.6 dB.

At frequencies further than ~150 MHz from the center frequency, the observed response differs significantly from the calculated response. This is at least partially due to the fact that the weights were calibrated at the center frequency of the passband, and, as is shown in Figure 4.2, the frequency response of the type of detector used in the transversal filter does not scale independently of frequency as the bias voltage is changed. Other possible contributions to the deviation from the calculated response are the inaccuracies in the tap weights and tap delay lengths, as discussed in the previous section.

The observed phase response of the bandpass filter with a center frequency at 350 MHz is shown in Figure 4.11 for the passband of the synthesized filter. As expected from equation (4.1), the phase response is a linear function of frequency. The variation in the

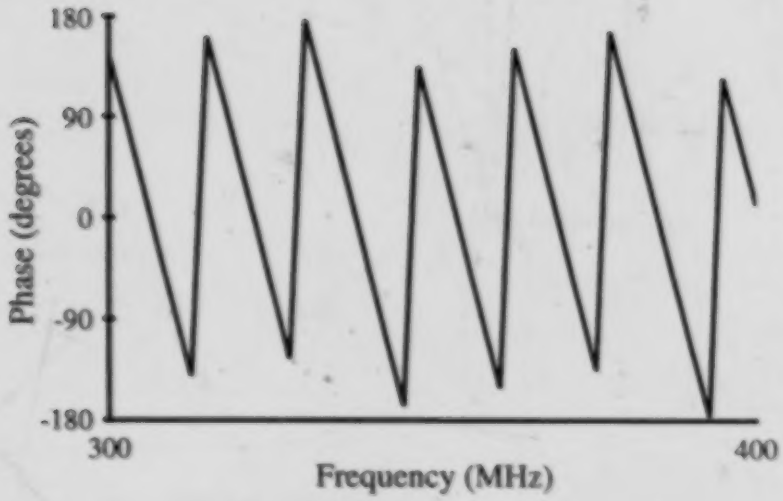

Figure 4.11. *Measured phase response of synthesized bandpass filter centered at 350 MHz*

wrap-around point is due to the limited number of data points which make up this plot: there are 5 or 6 data points evenly spaced in frequency for each 360° of phase shift.

The power response of a synthesized bandpass filter centered at 350 MHz is shown over the full 1 GHz bandwidth of the transversal filter in Figure 4.12. The presence of a passband centered at 650 MHz agrees with the form of the expected response shown in Figure 4.3.

To demonstrate the agility of the system, the controller was used to switch the transversal filter response between the bandpass filters centered at 200 MHz and 350 MHz shown in Figure 4.10. The system response to a 330 MHz sine wave during the transition was observed with a 350 MHz oscilloscope; a photograph of the event is shown in Figure 4.13. When the response was switched from the filter centered at 200 MHz to the one centered at 350 MHz, the amplitude of the sine wave went from 60 mV to 220 mV in 750 μs; this agrees with the reconfiguration time reported in Chapter 3. Note that the observed attenuation is only 11.3 dB whereas the difference in the response of the two filters as shown in Figure 4.10 is ~30 dB. This is because the data of Figure 4.10 were generated using an input filter bandwidth of 300 Hz while the sine wave in the transition time experiment was observed in the 350 MHz bandwidth of the oscilloscope. This illustrates the degradation in the transversal filter response due to the signal being near the noise level of the system.

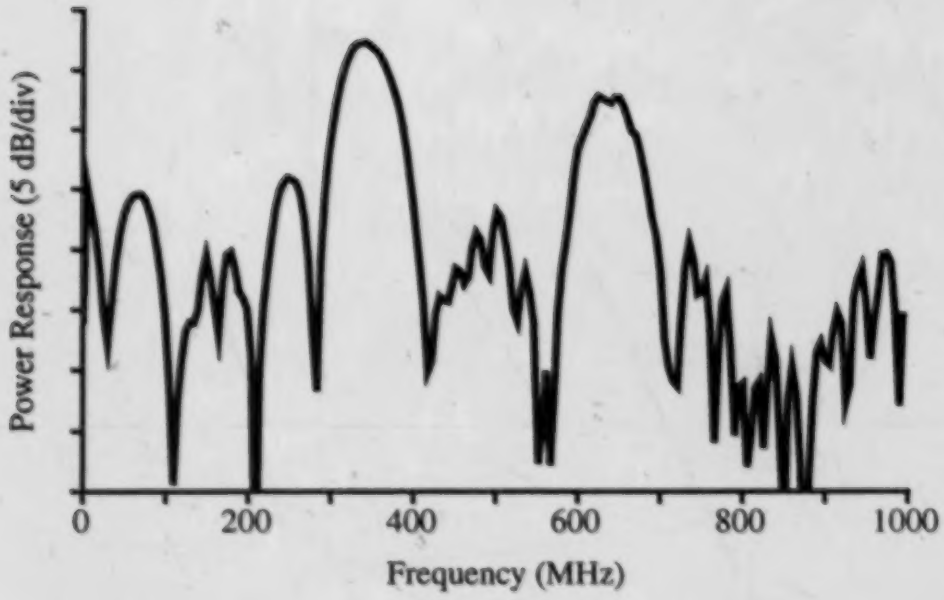

Figure 4.12. *Measured power response of synthesized bandpass filter centered at 350 MHz*

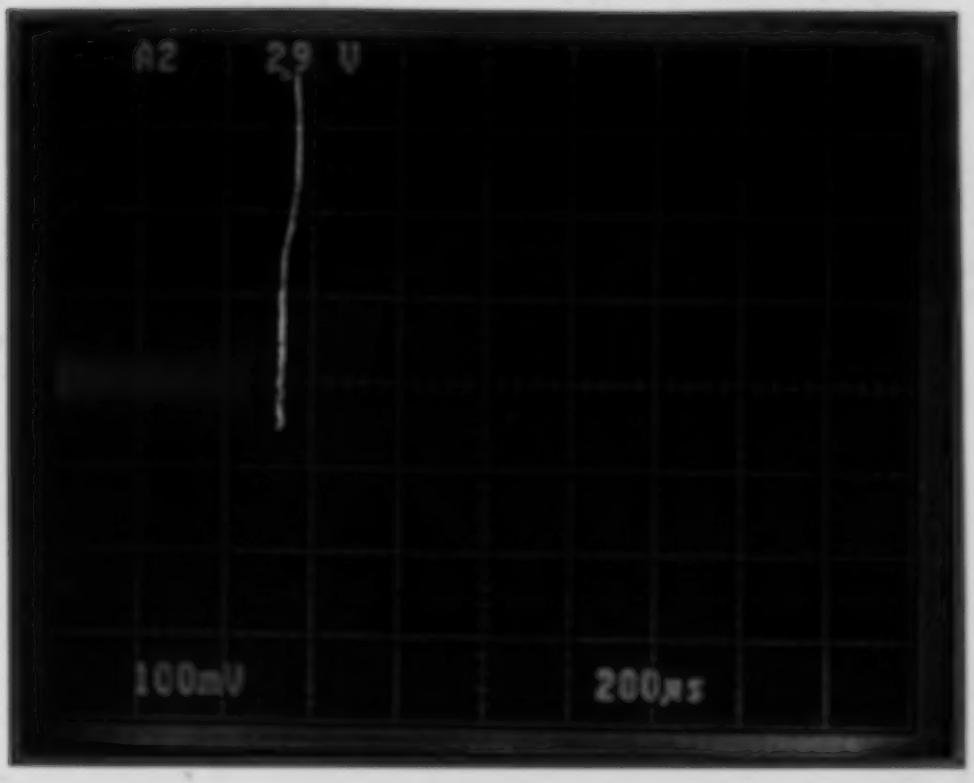

Figure 4.13. *System response to a 330 MHz sine wave during filter reconfiguration*

Chapter 5

Summary and Conclusions

This thesis has discussed a novel 16-tap transversal filter implementation which employs optical fiber delays and optoelectronically set weights. The filter and its experimental implementation are described also in reference [34], which is included as Appendix E. This device has the widest bandwidth of any transversal filter with more than eight controllable bipolar weights in the published literature. The technological problems involved in constructing the transversal filter, and their solutions, and experimental results were provided demonstrating the operation of the transversal filter. In addition, the design, construction, and operation were described of a multi-channel analog voltage controller which was built to facilitate the demonstration of the transversal filter. This controller can also be used in future work in optoelectronic signal processing and switching.

The novel feature of the transversal filter implementation described in this thesis is the use of optoelectronically set weights. By adjusting the dc bias voltage across an MSM photodetector we control its responsivity. This approach has advantages in simplicity, size, and cost over other weighting schemes. A characteristic of particular importance of this weighting technique is the abilility to implement bipolar weights by reversing the polarity of the detector bias which introduces a 180° phase shift in the output. Other weighting schemes require two channels for each delay path to implement bipolar weights.

The principal technological difficulties involved in the construction of the transversal filter were forming the optical delays and distributing the optical signals to the 1 X 16 linear array of MSM detectors. The delays were constructed by cutting optical fibers of the appropriate lengths and fusion splicing them into the delay paths. Fusion splices have the advantages of mechanical stability and low loss but the disadvantage of making it difficult to correct errors in the delay lengths.

The distribution of optical signals was achieved by pre-aligning the optical fibers in custom made V-grooves etched in silicon with the same center-to-center spacing as the detectors in the MSM array. After epoxying the fibers in the V-groove, the entire V-groove assembly is aligned with the detector array. With this technique optical coupling to the detectors of the array has been achieved with an observed uniformity of ±1.5 dB in electrical power.

In future work, improvements in filter performance will be achieved if a more reliable system of constructing precise fiber delays can be found. Integrated optics techniques offer potential for very precise delays, but at present could be used only for very broadband filters because of the constraints on length set by waveguide attenuation and fabrication limitations.

The bandwidth of filters of the type described in this thesis could be extended by using more advanced MSM detectors. TRLabs, SSOC, and CITR have recently completed a project to design MSM detectors for the 5 GHz range. Devices are expected to available for testing in 1992. As a result of the interest in MSM detectors for optoelectronic switching, signal processing, and for optical interconnects, a working group has been established to produce, by the end of 1992, controllable arrays of detectors with integrated pre-amplifiers.

To set the weights of the transversal filter, a multi-channel analog voltage controller has been constructed. It can control up to 100 voltages over a ±4.9 volt range with at least 0.2% accuracy. Individual weights can be reprogrammed in 50 μs.

To demonstrate the transversal filter, linear phase bandpass filters with a bandwidth of 50 MHz and center frequencies of 200 and 350 MHz were synthesized. Sidelobe rejections within 2 to 3 dB of the calculated value of 15.6 dB were obtained. Beyond the first sidelobe, the observed frequency response differed significantly from the calculated response. Simulation studies indicated that this is primarily due to errors in the tap delays; the 10% imprecision in the tap weights due to the weak sensitivity of the experimental detectors used also contributed to the degraded response.

The agility of the system was demonstrated by switching the synthesized response between the two bandpass filters and observing the system response to a sine wave near the center frequency of one of the passbands. The expected reconfiguration time of 750 μs was observed.

There are a number of possible avenues for future work. The work presented in this thesis could be extended to construct filters with a larger number of taps, a wider bandwidth, and greater tap weight and tap delay accuracy. The existing transversal filter could be used as part of an adaptive filter for real time wideband signal processing applications. Such filters will be of considerable importance in equalizing multi-path propagation for mobile radio. The concept of optoelectronically set weights and the technology used to implement the transversal filter could be applied to optical computing, specifically to the implementation of novel optoelectronic neural networks. In each of these applications, the existing multi-channel analog voltage controller would prove useful.

To conclude, an optoelectronic transversal filter with optical fiber delays, optoelectronically set bipolar weights, and a 1 GHz bandwidth has been successfully demonstrated. The active devices employed have the potential for operation up to about 10 GHz; with improvement in the precision of the passive delay lines, filters that are rapidly tunable over a very wide range are foreseeable. This work has applications in agile and adaptive wideband filters, equalizers, and neural networks.

References

1 H. E. Kallmann, "Transversal filters," *Proc. IEEE*, **28**, pp. 302-310, July 1940.

2 F. G. Stremler, Introduction to Communication Systems, Second Edition, Addison-Wesley Publishing Company, Reading, Massachusetts, 1982.

3 D. D. Buss, R. W. Brodersen, and C. R. Hewes, "Charge-coupled devices for analog signal processing," *Proc. IEEE*, **64**, pp. 801-804, May 1976.

4 R. A. Haken, R. C. Pettengill, and L. R. Hite, "A general purpose 1024-stage electronically programmable transversal filter," *IEEE J. Solid-State Circuits*, **SC-15**, pp. 984-996, Dec. 1980.

5 A. M. Chiang and B. E. Burke, "A high-speed digitally programmable CCD transversal filter," *IEEE J. Solid-State Circuits*, **SC-18**, pp. 745-753, Dec. 1983.

6 Y. S. Lee and K. W. Martin, "A switched-capacitor realization of multiple FIR filters on a single chip," *IEEE J. Solid-State Circuits*, **SC-23**, pp. 536-542, April 1988.

7 S. K. Sunter, E. F. Girczyc, and A. Chowaniec, "A programmable transversal filter for voice-frequency applications," *IEEE J. Solid-State Circuits*, **SC-16**, pp. 367-372, Aug. 1981.

8 J. B. Green, D. E. Oates, P. M. Grant, and D. L. Smythe, "Adaptive and matched filtering with a SAW/FET programmable transversal filter," in *1986 Ultrasonics Symp. Proc.*, pp. 137-141.

9 D. E. Oates, D. L. Smythe, J. B. Green, and R. S. Withers, "SAW/FET programmable filter with varistor taps for improved performance," in *1988 Ultrasonics Symp. Proc.*, pp. 155-158.

10 C. M. Panasik, W. R. Paxman, and J. W. Culver, "A 32-tap digitally controlled programmable transversal filter," in *1988 Ultrasonics Symp. Proc.*, pp. 151-154.

11 F. Guediri, R. L. Martin, B. J. Hunsinger, and F. M. Fliegel, "Performance of acoustic charge transport programmable tapped delay line," in *1987 Ultrasonics Symp. Proc.*, pp. 11-14.

12 R. W. Miller, C. A. Ricci, and R. J. Kansy, "An acoustic charge transport digitally programmable transversal filter," *IEEE J. Solid-State Circuits*, **SC-24**, pp. 1675-1682, Dec. 1989.

13 D. E. Cullen, W. J. Tanski, S. W. Merritt, R. N. Sacks, R. D. Carroll, and E. J. Branciforte, "Heterojunction acoustic charge transport device technology," in *1988 Ultrasonics Symp. Proc.*, pp. 135-143.

14 S. W. Merritt, W. J. Tanski, D. E. Cullen, R. N. Sacks, R. D. Carroll, and E. J. Branciforte, "A 3.35 microsecond HACT transversal filter," in *1989 Ultrasonics Symp. Proc.*, pp. 213-216.

15 R. S. Withers, "A comparison of superconductive and surface-acoustic-wave signal processing," in *1988 Ultrasonics Symp. Proc.*, pp. 185-194.

16 M. A. Delaney, R. S. Withers, A. C. Anderson, J. B. Green, and R. W. Mountain, "Superconductive delay line with integral MOSFET taps," *IEEE Trans. Magnetics*, **MAG-23**, pp. 791-795, March 1987.

17 C. T. Chang, J. A. Cassaboom, and H. F. Taylor, "Fibre-optic delay-line devices for r. f. signal processing," *Electron. Lett.*, **13**, pp. 678-680, Oct. 1977.

18 K. P. Jackson, J. E. Bowers, S. A. Newton, and C. C. Cutler, "Microbend optical fiber tapped delay line for gigahertz signal processing," *Appl. Phys. Lett.*, **41**, pp. 139-141, July 1982.

19 K. P. Jackson, S. A. Newton, and H. J. Shaw, "1 Gbit/s code generator and matched filter using an optical fiber tapped delay line," *Appl. Phys. Lett.*, **42**, pp. 556-558, April 1983.

20 S. A. Newton, K. P. Jackson, and H. J. Shaw, "Optical fiber V-groove transversal filter," *Appl. Phys. Lett.*, **43**, pp. 149-151, July 1983.

21 D. E. N. Davies and G. W. James, "Fibre-optic tapped delay line filter employing coherent optical processing," *Electron. Lett.*, **20**, pp. 95-97, Jan. 1984.

22 K. P. Jackson, Guoging Xiao, and H. J. Shaw, "Coherent optical fibre delay-line processor," *Electron. Lett.*, **22**, pp. 1335-7, Dec. 1986.

23 S. A. Pappert, C. T. Chang, and M. N. McLandrich, "Fiber optic transversal filters for rf direction finding," *Fiber and Integrated Opt.*, **6**, pp. 63-77, Jan. 1987.

24 C. C. Wang, "High-frequency narrow-band single-mode fiber-optic transversal filters," *J. Lightwave Technol.*, **LT-5**, pp. 77-81, Jan. 1987.

25 C. E. Lee, R. A. Atkins, and H. F. Taylor, "Reflectively tapped optical fibre transversal filters," *Electron. Lett.*, **23**, pp. 596-598, May 1987.

26 S. Gweon, C. E. Lee, and H. F. Taylor, "Wide-band fiber optic signal processor," *IEEE Photon. Technol. Lett.*, **1**, pp. 467-468, Dec. 1989.

27 D. M. Gookin and M. H. Berry, "Finite impulse response filter with large dynamic range and high sampling rate," *Appl. Opt.*, **29**, pp. 1061-1062, March 1990.

28 H. Yonezu, T. Himeno, K. Kanamori, K. Pak, and Y. Takano, "Optoelectronic synaptic connection circuit with variable analog and nonvolatile weights," in *Conf. Rec., International Topical Meeting on Optical Computing*, Kobe, Japan (Japan Society for Applied Physics, April 1990).

29 R. I. MacDonald and S. S. Lee, "Photodetector sensitivity control for weight setting in optoelectronic neural networks," *Appl. Opt.*, **30**, pp. 176-179, Jan. 1991.

30 Y. Nitta, J. Ohta, S. Tai, and K. Kyuma, "Variable-sensitivity photodetector that uses a metal-semiconductor-metal structure for optical neural networks," *Opt. Lett.*, **16**, pp. 611-613, April 1991.

31 K. Jacobson, "Optical Time-of-Flight Multiplexing," MSc. Thesis, University of Alberta, 1990.

32 S. Paranjpe, "Controller for a 10 x 10 Opto-Electronic Switch Matrix," TRLabs Internal Report, April 1990.

33 M. Veillieux, "Construction of a 10x10 Broadband Matrix Switch," MSc. Thesis, University of Alberta, 1990.

34 B. E. Swekla and R. I. MacDonald, "Optoelectronic transversal filter," *Electron. Lett.*, **27**, pp. 1769-1770, Sept. 1991.

35 K. Sasayama, M. Okuno, and K. Habara, "Coherent optical transversal filter using silica-based single-mode waveguides," *Electron. Lett.*, **25**, pp. 1508-1509, Oct. 1989.

36 K. Sasayama, M. Okuno, and K. Habara, "Coherent optical transversal filter using silica-based waveguides for high-speed signal processing," *J. Lightwave Technol.*, **LT-9**, pp. 1225-1230, Oct. 1991.

Appendix A

Printed Circuit Board Masks

and

Component Placement Diagrams

This appendix contains the printed circuit board masks used in the construction of the optoelectronic transversal filter receiver boards, the D/A converter board, and the address decoding and sample-and-hold circuit board. All circuit board masks are plotted at their actual size. Component placement diagrams are also given for the D/A converter board and the address decoding and sample-and-hold circuit board.

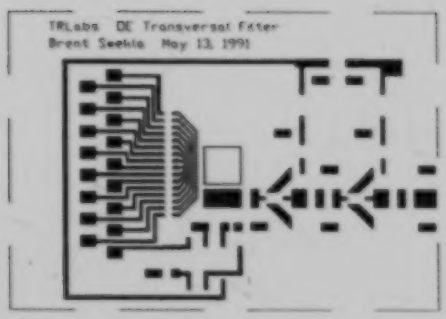

Figure A.1 (a) *Optoelectronic transversal filter receiver: Board 1 mask*

Figure A.1 (b) *Optoelectronic transversal filter receiver: Board 2 mask*

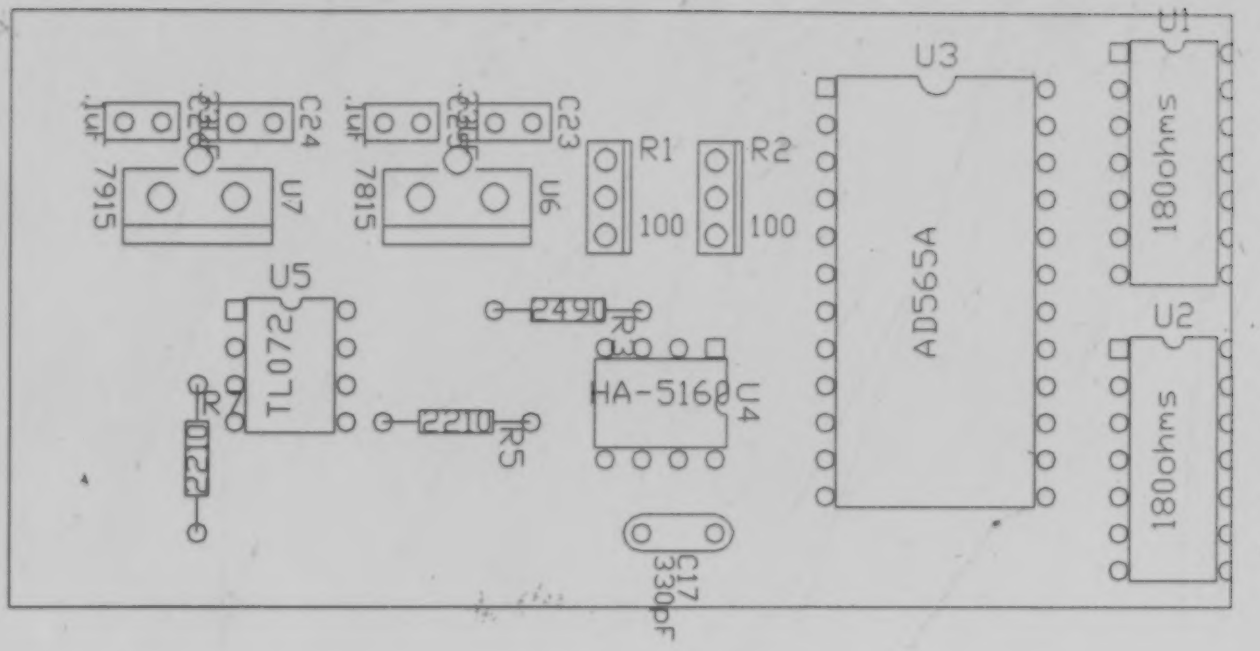

Figure A.2 (a) *D/A converter board: Component placement diagram - top side*

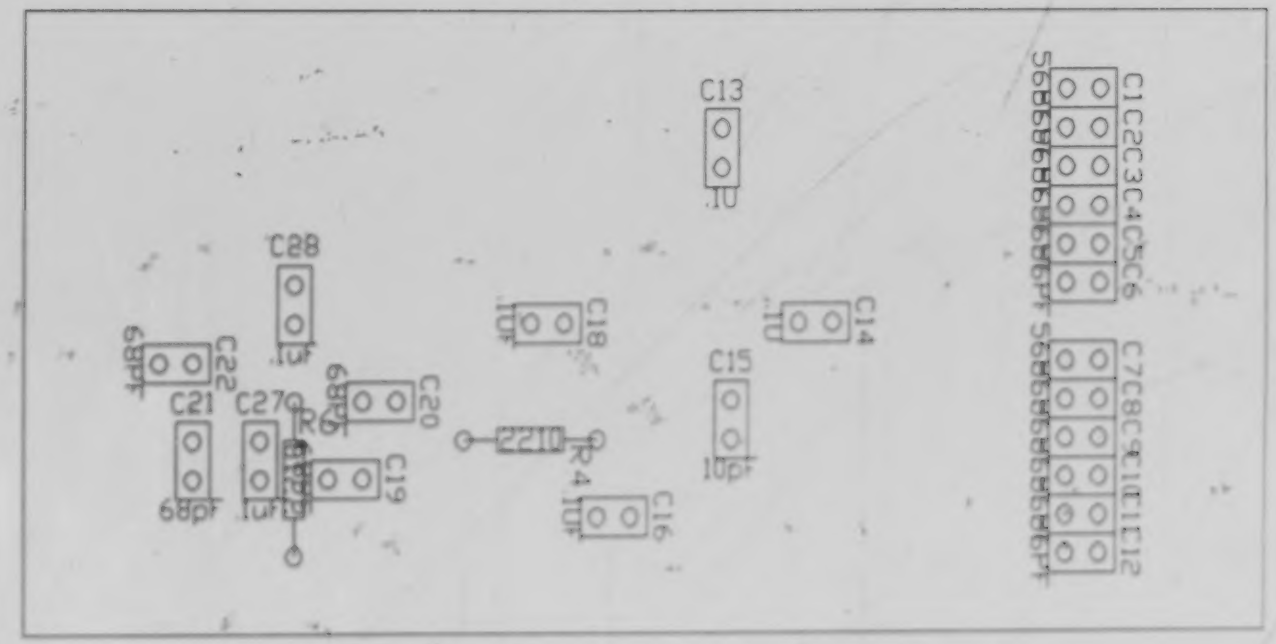

Figure A.2 (b) *D/A converter board: Component placement diagram - bottom side*

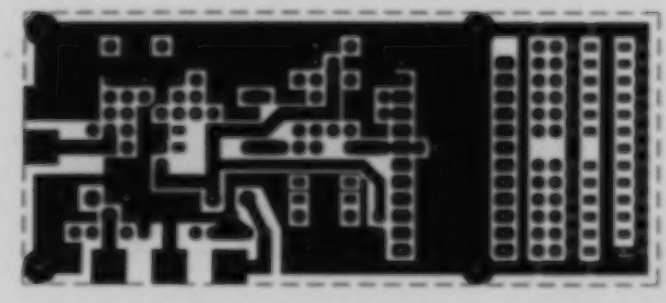

Figure A.3 (a) *D/A converter board: Printed circuit board mask - top side*

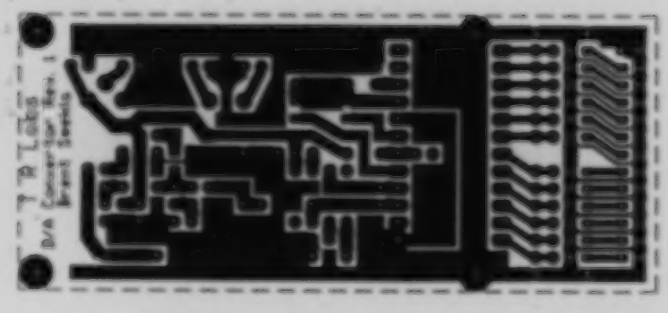

Figure A.3 (b) *D/A converter board: Printed circuit board mask - bottom side*

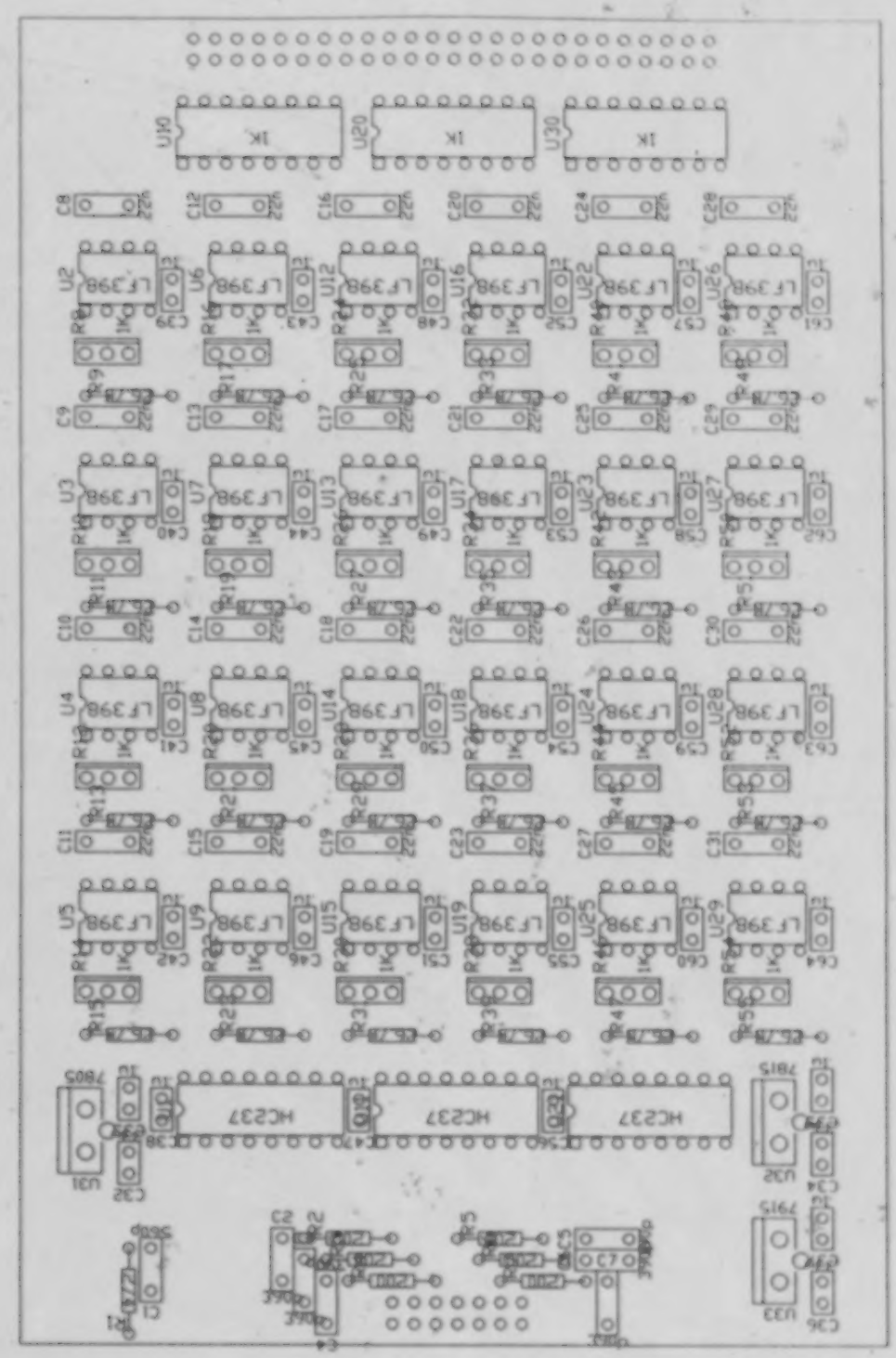

Figure A.4. *Address decoding and sample-and-hold circuit board: component placement diagram*

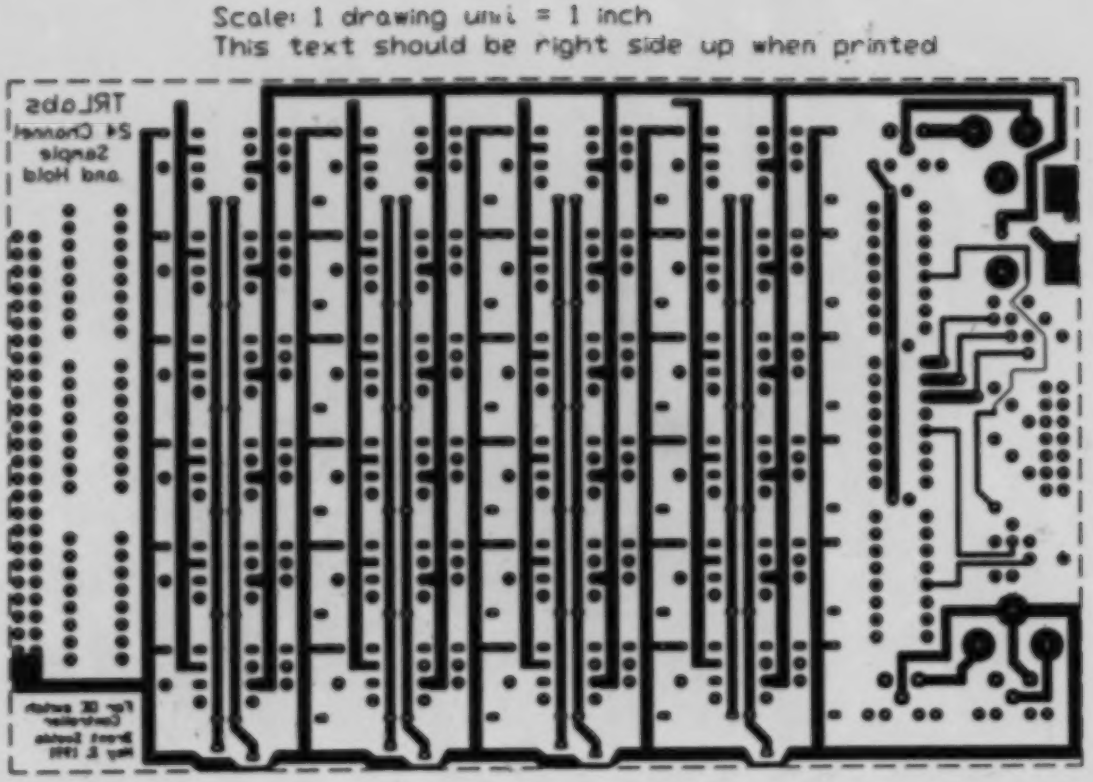

Figure A.5 (a) *Address decoding and sample-and-hold circuit board: Printed circuit board mask - top side*

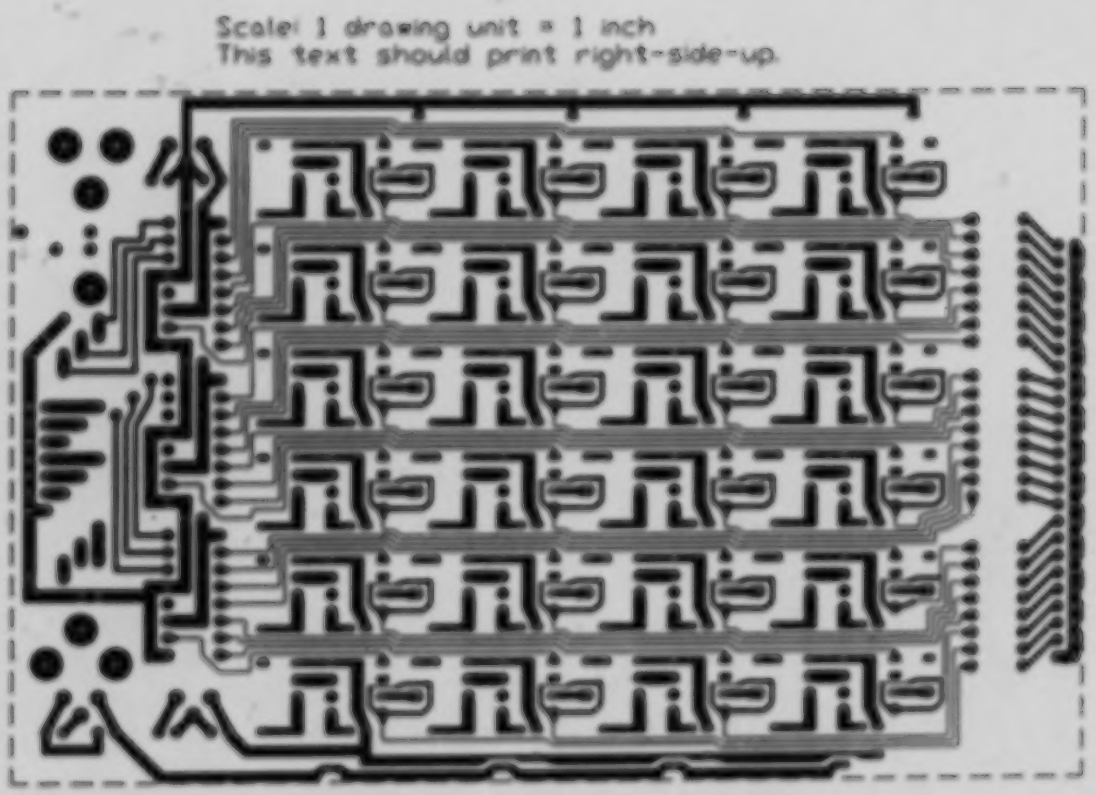

Figure A.5 (b) *Address decoding and sample-and-hold circuit board: Printed circuit board mask - bottom side*

Appendix B

Program Listings

This appendix contains the text of the user interface and 68HC11 microcontroller software.

B.1 User Interface Software

B.1.1 PORTDEFN.H

```
/*************************************************************************
        This file contains some definitions and macros which make the
        idiosyncracies of the IBM printer adapter and parallel port
        transparent to the Multi-Channel Analog Voltage
        Controller programmer.
*************************************************************************/
/* PARALLEL PORT LOCATION
        The 25 pins of the port correspond to three locations in
        memory starting at BASE which is machine dependent. The pins
        are not in order and pins 18-25 are grounded.
*/
#define  BASE                   0x378
#define  DATA_OUT_PORT          BASE
#define  DATA_IN_PORT           BASE + 1
#define  CONTROL_PORT           BASE + 2

/* PORT MASKS
        Some of the pins are inverted by the printer adapter. They
        can be "uninverted" by XORing the "raw" port values with the
        following masks.
*/
#define  DATA_IN_MASK           0x80
#define  CONTROL_MASK           0x0A

/* COUNTER COMMANDS
        These are commands which can be sent to the counter in the
        controller. Load and Reset on the 74HC161 are active-low.
 CLEAR sets the counter output to 0000 (Reset pin on the
        counter).
 LOAD sets the counter output to the preset value (1111)
        on the next rising edge of the clock (STROBE).
RESET sets the load and reset pins on the counter high.
        This must be done after a LOAD or a CLEAR command to
        enable the counter. RESET also sets the clock low.
*/
#define  CLEAR                  0x0C
```

```
#define  LOAD                    0x0A
#define  RESET                   0x0F

/* STROBE CONTROL
        The following definitions are for a command which strobes the
        clock pin on the counter without disturbing the other control
        lines. The strobe line is also inverted by the printer
        adapter. The extra writes to the parallel port are due to the
        speed of this instruction and the pcjk slow rise time. The
        strobe pin is actually connected to the counter through a
        Schmitt trigger, so if the strobe line is set high and then
        low on successive writes the trigger threshold is not reliably
        reached. As defined STROBE_COUNTER produces a reliable pulse
        between 2 and 3 microseconds long on a PC with a 10 MHz clock.
*/
#define  STROBE                  0x01
#define  LO                      (CONTROL_REGISTER | STROBE)
#define  HI                      (CONTROL_REGISTER &~STROBE)
#define  STROBE_COUNTER          outportb(CONTROL_PORT,HI); \
                                 outportb(CONTROL_PORT,HI); \
                                 outportb(CONTROL_PORT,HI); \
                                 outportb(CONTROL_PORT,LO); \
                                 outportb(CONTROL_PORT,LO); \
                                 outportb(CONTROL_PORT,LO)

/* MACRO COMMANDS
        The following macro definitions perform the port I/O
        transparent to the programmer.
*/
#define  SEND_COMMAND(y)         CONTROL_REGISTER &= STROBE; \
                                 CONTROL_REGISTER |= (y ^ CONTROL_-
MASK); \
                                 outportb(CONTROL_PORT, CONTROL_REGIS-
TER)
#define  SEND_DATA(y)            outportb(DATA_OUT_PORT, y)
#define  GET_DATA(y)             y=(inportb(DATA_IN_PORT)&0x0F)^DATA_IN_-
MASK
```

B.1.2 IBM_OETF.C

```
/*****************************************************************************
IBM_OETF.C
 This program controls the 16 tap OETF. It contains a number
 of functions which can be used to demonstrate and test the opto-
 electronic transversal filter.
*****************************************************************************/
#include <stdio.h>
#include <dos.h>
#include <D:\OECONTRL\PC_PRGM\portdefn.h>

#define BUSY !NOT_BUSY
#define UPDATE 1
#define REQUEST 2
#define DATA1 4
#define DATA2 5
#define DATA3                   6
#define NOT_BUSY_MASK 0x08
#define ESC 0x1B

int cp = 3, bias_value = 2047, delta = 25;
int bias1 = 2047, bias2 = 4095, tdelay = 50;
int i, j, error, command = UPDATE;
int CONTROL_REGISTER, NOT_BUSY;
double bias_voltage;

main()
{
      int choice;
      do {
            clrscr();
            SEND_COMMAND(LOAD);
            SEND_COMMAND(RESET);
            printf(" OPTOELECTRONIC MATRIX SWITCH CONTROLLER\n\n\n");
            printf(" 1: Set a specific crosspoint's bias\n");
            printf(" 2: Increment/Decrement bias\n");
            printf(" 3: Alternate bias between two values\n");
            printf(" 4: Set all crosspoint biases to one value\n");
            printf(" 5: Set all biases alternately to two values\n");
```

```
printf(" 6: Uniformly space all biases between two values\n");
printf(" 7: Select data set 1\n");
printf(" 8: Select data set 2\n");
printf(" 9: Select data set 3\n");
printf("\n\n Try yer luck .... or press ESC to exit");
choice = getch();
switch (choice) {
        case '1':
                set_bias();
                break;
        case'2':
                increment_bias();
                break;
        case '3':
                switch_bias();
                break;
        case '4':
                set_all();
                break;
        case '5':
                alternate_all();
                break;
        case '6':
                ramp_all();
                break;
        case '7':
                command = DATA1;
                write_data();
                command = UPDATE;
                break;
        case '8':
                command = DATA2;
                write_data();
                command = UPDATE;
                break;
        case '9':
                command = DATA3;
                write_data();
                command = UPDATE;
```

```
                    break;
                default:
                    ;
            }
        }
        while (choice != ESC);
}

set_bias()
{
        clrscr();
        error = 1;
        do {
            printf("\n Enter crosspoint address (0 - 99): ");
            scanf("%d", &cp);
            printf("\n Enter bias value (0 - 4095): ");
            scanf("%d", &bias_value);
            if ((cp<0) || (cp>99) || (bias_value<0) || (bias_value>4095))
                printf("\n Error Error Error Error Errrrrrrrrrrror\n\n");
            else
                error = 0;
        }
        while (error);
        write_data();
}

increment_bias()
{
        int choice;
        increment_menu(delta);
        write_data();
        do {
            choice = getch();
            switch(choice) {
                case 'i'
                    bias_value += delta;
                    if (bias_value > 4095)
                        bias_value = 4095;
                    bias_voltage = -5. + bias_value * 10./4096.;
```

```
                printf("\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b"
                    "%-4d (%-+7.4f Volts)", bias_value, bias_voltage);
                write_data();
                break;
            case 'd':
                bias_value -= delta;
                if (bias_value < 0)
                    bias_value = 0;
                bias_voltage = -5. + bias_value * 10./4096.;
                printf("\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b\b"
                    "%-4d (%-+7.4f Volts)", bias_value, bias_voltage);
                write_data();
                break;
            case 'c':
                clrscr();
                error = 1;
                do {
                    printf("\n Enter new increment: ");
                    scanf("%d", &delta);
                    if ((delta <= 0) || (delta > 4095))
                        printf("\n Error Error Errrrrrrrrrrrrrror\n\n");
                    else
                        error = 0;
                }
                while (error);
                increment_menu();
                break;
            }
        }
        while (choice != ESC);
}

increment_menu()
{
        clrscr();
        printf("\n Press i to increment\n"
            " d to decrement\n"
            " c to change increment\n"
            " ESC to exit\n\n");
```

```
        bias_voltage = -5. + bias_value * 10./4096.;
        printf(" The current crosspoint is %d\n"
               " The increment is set at %d\n"
               " The crosspoint bias is %-4d (%-+7.4f Volts)",
               cp, delta, bias_value, bias_voltage);
}

switch_bias()
{
        int choice, k;
        do {
                switch_menu(bias1, bias2, tdelay);
                do {
                        bias_value = bias1;
                        write_data();
                        for (k = 0 ; k < tdelay ; k++) {
                                for (j = 0 ; j < tdelay ; j++)
                                        ;
                        }
                        bias_value = bias2;
                        write_data();
                        for (k = 0 ; k < tdelay ; k++) {
                                for (j = 0 ; j < tdelay ; j++)
                                        ;
                        }
                }
                while ( !kbhit() );
                choice = getch();
                switch(choice) {
                        case 't':
                                clrscr();
                                error = 1;
                                do {
                                        printf("\n Enter new time delay factor: ");
                                        scanf("%d", &tdelay);
                                        if ( (tdelay < 0) || (tdelay > 32000) )
                                                printf("\n\n Error Error Errrrrrrrrrror\n\n\n");
                                        else
                                                error = 0;
```

```
                    }
                    while (error);
                    break;
                case 'b':
                    clrscr();
                    error = 1;
                    do {
                        printf("\n Enter first bias value (0 - 4095): ");
                        scanf("%d", &bias1);
                        printf("\n Enter second bias value (0 - 4095): ");
                        scanf("%d", &bias2);
                        if ((bias1<0)||(bias1>4095)||(bias2<0)||(bias2>4095))
                            printf("\n Error Error Error Errrrrrrrrrrrrror\n\n");
                        else
                            error = 0;
                    }
                    while (error);
                    break;
            }
        }
        while (choice != ESC);
}

switch_menu()
{
    clrscr();
    printf("\n Press t to change time delay"
        "\n b to change bias levels"
        "\n ESC to exit\n\n"
        "\n The current crosspoint is %d"
        "\n The first bias level is %-4d (%-+7.4f Volts)"
        "\n The second bias level is %-4d (%-+7.4f Volts)"
        "\n The current time delay factor is %-8d", cp,
        bias1, -5. + bias1 * 10. / 4096.,
        bias2, -5. + bias2 * 10. / 4096., tdelay);
}

set_all()
{
```

```
        clrscr();
        error = 1;
        do {
                printf("\n Enter global bias value (0 - 4095): ");
                scanf("%d", &bias_value);
                if ( (bias_value < 0) || (bias_value > 4095) )
                        printf("\n\n Error Error Errrrrrrrrror!\n\n\n");
                else
                        error = 0;
        }
        while (error);
        j = cp;
        for (cp = 0 ; cp < 100 ; cp++)
                write_data();
        cp = j;
}

alternate_all()
{
        clrscr();
        error = 1;
        do {
                printf("\n Enter first bias value (0 - 4095): ");
                scanf("%d", &bias1);
                printf("\n Enter second bias value (0 - 4095): ");
                scanf("%d", &bias2);
                if ((bias1<0)||(bias1>4095)||(bias2<0)||(bias2>4095))
                        printf("\n Error Error Error Errrrrrrrrrrrror\n\n");
                else
                        error = 0;
        }
        while (error);
        j = cp;
        for (i = 0 ; i <= 98 ; i += 2) {
                cp = i;
                bias_value = bias1;
                write_data();
                cp++;
                bias_value = bias2;
```

```
            write_data();
        }
        cp = j;
}

ramp_all()
{
        clrscr();
        error = 1;
        do {
            printf("\n Enter first bias value (0 - 4095): ");
            scanf("%d", &bias1);
            printf("\n Enter second bias value (0 - 4095): ");
            scanf("%d", &bias2);
            if ((bias1<0)||(bias1>4095)||(bias2<0)||(bias2>4095))
                printf("\n Error Error Error Errrrrrrrrrrrrror\n\n");
            else
                error = 0;
        }
        while (error);
        j = cp;
        bias_value = bias1;
        for (cp = 0; cp < 100 ; cp++) {
            write_data();
            bias_value += (bias2 - bias1) / 99.;
        }
}

write_data()
{
        short bias_low, bias_high;
        bias_low = (bias_value & 0x00FF);
        bias_high = (bias_value >> 8);
        SEND_DATA(command);
        SEND_COMMAND(CLEAR);
        SEND_COMMAND(RESET);
        SEND_DATA(cp);
        STROBE_COUNTER;
        SEND_DATA(bias_high);
```

```
        STROBE_COUNTER;
        SEND_DATA(bias_low);
        STROBE_COUNTER;
        GET_DATA(NOT_BUSY);
        NOT_BUSY &= NOT_BUSY_MASK;
        if (BUSY) {
            printf("\n 68HC11 servicing an interrupt -- TRANSFER INTO ARTS!!");
            exit(0);
        }
        j = 0;
        STROBE_COUNTER;
        STROBE_COUNTER;
        GET_DATA(NOT_BUSY);
        NOT_BUSY &= NOT_BUSY_MASK;
        while (BUSY) {
            GET_DATA(NOT_BUSY);
            NOT_BUSY &= NOT_BUSY_MASK;
            if (j > 1000) {
                printf("\n 68HC11 runaway -- HAH HAH HO HEE HOO (slap)!!");
                exit(0);
            }
            j++;
        }
}
```

B.2 68HC11 Microcontroller Software

B.2.1 intial.doc

```
* Initial sequence: initializes uP registers, etc.
ORG      $C000                 set PC to top of EVB user RAM
INITIAL  LDAB   #OUTMASK       set portD as output
         STAB   DDRD
         LDAA   #GLOBAL        disable all devices
         STAA   PORTD          put command on portD
         STAA   PORTB          dummy write to pulse STRB
         LDAA   #SQCLK         set QCLK as Q=1
         STAA   PORTD          put command on portD
         STAA   PORTB          dummy write to pulse STRB
         LDAA   #NOTBUSY       tell PC that uP
         STAA   PORTD          put command on portD
         STAA   PORTB          dummy write to pulse STRB
         LDAA   #WASTE         set ports as Hi-Z inputs
         STAA   PORTD          portD = Hi-Z input
         LDAB   #OUTMASK       set ports as outputs
         STAB   DDRC           portC = output
         LDAB   #$06           set simple strobe mode
         STAB   PIOC
* set up JMP locations for XIRQ and IRQ interrupt vectors
         LDX    #INTERRUPT     addr of int. seq for xirq
         STX    $00F2          store xirq int. vector
         LDA    #$7E           load jump command
         STA    $00F1
         LDX    #READDATA      addr. of readdata sequence
         STX    $00EF          store IRQ int. vector
         STA    $00EE
         LDAA   #$00           clear the CCR
         TAP                   X bit=Ibit=0
         JMP    REFRESH        start to cycle xpoint data
```

B.2.2 declare.doc

```
* Declaration section of constants to be used in program
* Names of ports instead of using their address numbers
PORTB    EQU    $1004
```

```
PORTC    EQU    $1003
PORTCL   EQU    $1005
PORTD    EQU    $1008
PIOC     EQU    $1002

* Names of data direction registers and masks
DDRC     EQU    $1007
DDRD     EQU    $1009
INMASK   EQU    $00
OUTMASK  EQU    $FF

* Names of 4 by 16 control signal decoder lines
* all values are in decimal
ONDALO   EQU    $02      enables output of DATALO data latch
ONDAHI   EQU    $06      enables output of DATAHI data latch
ONADDR   EQU    $0A      enables output of ADDR data latch
ONCOMM   EQU    $0E      enables output of COMMANDdata latch
ONPCDA   EQU    $12      data can be clocked into PCDATA
OFFPCDA  EQU    $16      data not clocked into PCDATA latch
RQCLK    EQU    $1A      resets QCLK FF, alternates clocking
ONDA     EQU    $1E      enables output of D/A latches
ONSEQ    EQU    $22      enables CS1 of 4x16 SEQ decoder
SQCLK    EQU    $26      sets QCLK FF, no clocking of D/A
OFFDA    EQU    $2A      disable D/A latch outputs
OFFSEQ   EQU    $2E      disable 4x16 SEQ decoder
WASTE    EQU              $32 unused BUSIO control line
NOTBUSY  EQU    $3A      tells PC that int. seq. is complete
GLOBAL   EQU    $3E      hi-z output latches, no clocking

* Names of const.and var used in Interrupt sequence
INXIRQ   FCB    $00      indicates if processing interrupt
INLOAD   FCB    $00
IMPCOUNT FCB    4        counter for signalling BusIO decode
COUNT    FCB    0        counter for indexing data into RAM
INFOZONE    FDB    $0000    loc. of where to load data
DATAZONE    FDB    $C700    addr. of where to load xpoint data
DATZONE1    EQU    $C700
DATZONE2    EQU    $C900
DATZONE3    EQU    $CB00
```

```
NEWZONE     EQU   $C500          addr. of where to load new data
NEWCOMM     EQU   NEWZONE+ONCOMM     location of COMMAND in ram
NEWADDR     EQU   NEWZONE+ONADDR     location of ADDR in ram
NEWDAHI     EQU   NEWZONE+ONDAH      location of DATAHI in ram
NEWDALO     EQU   NEWZONE+ONDALO     location of DATALO in ram

UPDATE      EQU   $01          command to start update sequence
REQUEST     EQU   $02          command to start request sequence
LOAD        EQU   $03          command to start loaddata sequence
DATA1       EQU   $04
DATA2       EQU   $05
DATA3       EQU   $06

* Names of const.and var used in Refresh sequence
ADDR        FCB   198          counter used to access xpoint DATA
MAX         EQU   198          (198/2) xpoints, starting at 0

* Names of const.and var used in Update sequence
TEMPADDR    FCB   $00          stores uP's version of ADDR
```

B.2.3 refresh.doc

```
* Refresh cycle
* PRE: simple strobe mode, active low pulse;
* PORTD is Hi-Z input; all dev. disabled
* All crosspoint data in mem; STRA interrupt disabled

* Enable Devices
* use acc A to store BusIO commands; acc.B<ADDR
REFRESH  LDAA  #OUTMASK     put output mask into Breg
 STAA    DDRD               make portD output port
 LDAA    #RQCLK             allows alternate clocking
 STAA    PORTD
 STAA    PORTB              Dummy write pulses STRB
         LDAA  #0
         CMPA  INXIRQ
         BEQ   REFCONT
         LDAA  #7
INSLOOP  INS
         DECA
```

```
            BNE     INSLOOP
            PULX
            CPX     #QCLKH1
            BLT     SETSP
            CPX     #QCLKH2
            BGT     SETSP
            LDAA    #SQCLK
            STAA    PORTD
            STAA    PORTB
SETSP       PSHX
            LDAA    #7
DESLOOP     DES
            DECA
            BNE     DESLOOP
REFCONT     LDAA    #ONDA       enable D/A latch output
            STAA    PORTD       put command on portD
            LDAB    ADDR        get xpoint addr (counter)
            LDY     DATAZONE    get start loc of xpoint data
            ABY                 set up dynamic index offset
            LDX     $00,Y       load x reg w/value at loc.
            STX     PORTC       dble. byte write to ports C&B
* data is on bus; STRB should have been pulsed
* by above command
            LDAA    #ONSEQ      enable 4x16 SEQ decoder
            STAA    PORTD       put command on portD
            LSRB                divide ADDR by 2 for actual ADDR
            STAB    PORTB       STRB is pulsed, ADDR on portB
            LDAA    #WASTE      change portD to Hi-Z inputs
            STAA    PORTD
            LDAA    #$00        set up compare with zero
            CMPA    INXIRQ      if interrupt in progress
            BEQ     NOTLONG     check if not a long branch
            JMP     RETINT      return to interrupt seq.
NOTLONG     EQU                 *notlong = program counter
* Refresh next crosspoint
            LDAB    ADDR        put (ADDR) into acc. A
            JMP     START
RESTART     LDAB    #MAX        Max number of crosspoints
            STAB    ADDR        re-initialize ADDR
```

```
START    LDY    DATAZONE      start loc. of xpoint data
         LDAB   ADDR
         ABY                  set up dynamic index offset
         LDX    $00,Y         dble. byte read loc. (ADDR)
         STX    PORTC         dble. byte write to ports C&B
* DATA clocked into D/A latch
QCLKH1   TBA                  copy acc B to acc A
         LSRA                 divide ADDR by 2 for addr
QCLKH2   STAA   PORTB         send D/A's value to approp. demux
         LDX    #12
DELAY    DEX
         BNE    DELAY
         CMPA   #0
         BEQ    RESTART
         SUBB   #2            subtract 2 from ADDR
         STAB   ADDR          restore new value of ADDR
         JMP    START
```

B.2.4 intseq.doc

```
* Interrupt sequence
* PRE: simple strobe mode, active low pulse;portD Hi-Z input
* STRA interrupt disabled; X Bit=I bit=0
* INXIRQ=INLOAD= $00
* POST: Same as PRE

* All Devices put on Hi-Z or disabled
INTERRUPT LDAB   #GLOBAL       Globally disable all devices
          STAB   PORTD         put command on portD
          LDAA   #OUTMASK      configure portD as output port
          STAA   DDRD
          LDAB   ADDR          prevent glitch on dummy write
          LSRB                 ensure proper address
          STAB   PORTB         dummy write to pulse STRB
          STAA   INXIRQ        interrupt service in prog.

* Allow for nested interrupts
* try a dummy STRB in simple strobe mode, and change to
* full input handshake inside READDATA
          LDAB   #$56          change to full inputhandshake
```

```
                STAB    PIOC            write to the I/O control reg.
                LDAB    #INMASK         change portC to input port
                STAB    DDRC            write to port control reg.
                CLI                     enables further int. from STRA

* Enable next latch device
                LDX     #NEWZONE        index data new data into ram
                STX     INFOZONE
                LDAA    #$0E            initialize impcount to 4
NEXTREAD        STAA    IMPCOUNT        store value back into memory
                STAA    PORTD           put impcount on portD
                STAA    COUNT           store value into COUNT
                LDAB    PORTCL          dummy read to trigger STRB
                NOP                     kill time
*               STAB    PORTB           dummy write to trigger STRB
*               LDAB    #$D6            force a STAF,STAI
*               STAB    PIOC
*               LDAB    PORTCL          dummy read to trigger STRB
* Interrupt from STRA should be triggered
* move READDATA to a different location !!! see straint.doc
* which contains code for READDATA
* finished reading yet?
                LDAA    IMPCOUNT        check impcounter
                SUBA    #$04
                BGT     NEXTREAD        if impcounter=0 then NEXTREAD
                LDAB    #$06            change I/0 to simple strobe
                STAB    PIOC
                LDAB    #OUTMASK        change portC to output port
                STAB    DDRC
                SEI                     disallow further IRQ ints

* Execute command (this is a jump table for subroutines)
                LDAA    NEWCOMM         what is the interrupt command?
                CMPA    #UPDATE         if UPDATE is requested then
                BNE     AROUND1         allows for long branches
                JMP     UPDSEQ          goto update sequence
AROUND1         EQU     *               around1 equivalent to PC
                CMPA    #REQUEST        if REQUEST is requested then
                BNE     AROUND2         allows for long branches
```

```
            JMP     REQSEQ          goto request sequence
AROUND2     EQU     *               around2 equivalent to PC
            CMPA    #LOAD           if LOAD is requested then
            BNE     AROUND3         allows for long branches
            JMP     LOADDATA        goto loaddata sequence
AROUND3     EQU     *               around3 equivalent to PC
            CMPA    #DATA1
            BNE     AROUND4
            PSHX
            LDX     #DATZONE1
            STX     DATAZONE
            PULX
            JMP     RETURN1
AROUND4     CMPA    #DATA2
            BNE     AROUND5
            PSHX
            LDX     #DATZONE2
            STX     DATAZONE
            PULX
            JMP     RETURN1
AROUND5     CMPA    #DATA3
            BNE     AROUND6
            PSHX
            LDX     #DATZONE3
            STX     DATAZONE
            PULX
            JMP     RETURN1
AROUND6     JMP     ERROR           if unrecognized command then
* goto error sequence

* Tell IBM PC that no longer busy, and ensure uP same
* mode as when entered
RETURN1     LDAB    #$06
            STAB    PIOC            ensure simple I/O mode
            LDAB    #NOTBUSY        signal IBM PC
            STAB    PORTD           put command on portD
            STAB    PORTB           dummy write to pulse STRB
            LDAB    #WASTE          configure portD Hi-Z inputs
            STAB    PORTD
```

```
                LDAB    #40
                STAB    ADDR

* Prepare to return from XIRQ interrupt
                JMP     REFRESH         ensure D/A and 4x16 are active
RETINT          LDAA    #$00            clear INXIRQ;no longer in int.
                STAA    INXIRQ
                RTI                     return to refresh
```

B.2.5 straint.doc

```
* Straint sequence; this code is executed when a STRA
* interrupt is serviced
* move READDATA to a different location !!!
READDATA        LDAB    #WASTE          configure portD as Hi-Z input
                STAB    PORTD
                LDX     INFOZONE        ram location for indexing
                LDAB    PIOC            set up read
                LDAA    PORTCL          get data from latch into reg A
                LDAB    COUNT           set up dynamic indexing offset
                ABX                     add COUNT to X register
                STAA    $00,X           put data into ram by indexing
* if never config. portD as Hi-Z, donàt need to reconfig.
*               LDAB    #OUTMASK        configure portD as output
*               STAB    DDRD            allows for next device enable
                RTI
```

B.2.6 updseq.doc

```
* Update sequence
* PRE: Simple strobe mode, active low pulse;
* portD is output; all dev. disabled
* POST: same as pre, but new DATA is put onto D/A latches

UPDSEQ          PSHA    save registers
                PSHB
                PSHX
                PSHY
* Enable devices, use acc B for BusIO commands
                LDAA    NEWDAHI         get hi data byte
                LDAB    NEWDALO         get lo data byte
```

```
            XGDX                    put data into X register
            LDAA    NEWADDR         get addr. of xpoint to update
            LDAB    #RQCLK          allows altenating clock
            STAB    PORTD
            STAA    PORTB           dummy write pulses STRB
            LDAB    #ONDA           enable D/A latch output
* PD0 & PD1 of port D are burned to 0 & 1 respectively
* a double byte read is not possible since the lo and hi
* data bytes are not in successive locations; load the data
* into acc. A & B then exchange with X register, as above
*           LDX     NEWDALO         dble. byte read of new DATA
            STAB    PORTD           put command on portD
            STX     PORTC           dble. byte write
            TAB                     copy acc A to acc B
            ASLB                    get uP's version of data addr.
            STAB    TEMPADDR        save uP's vers. of data addr
            LDY     DATAZONE        prepare for saving data
            ABY                     set up dynamic index offset
            STX     $00,Y           save new data in mem
            LDAB    #ONSEQ          enable 4x16 SEQ decoder
            STAB    PORTD           put command on portD
            STAA    PORTB           addr. of xpoint on B, STRB pls
            LDY     #12
DELAY2      DEY
            BNE     DELAY2
* Global disable
            LDAB    #GLOBAL         disable all devices
            STAB    PORTD           put command on portD
            STX     PORTC           dummy write to pulse STRB
            PULA                    restore registers
            PULB
            PULX
            PULY
            JMP     RETURN1         go back to interrupt sequence
```

B.2.7 reqseq.doc

```
* Request sequence
* PRE: Simple strobe, active low pulse; portD output
* all dev. Hi-Z;
```

```
* POST: Same as pre, but DATA is stored into INTFC latches
REQSEQ  PSHA                    save registers
        PSHB
        PSHX
        PSHY
* Enable latches
        LDAA  #ONPCDA           enable latch clocking
        LDAB  NEWADDR           get location of data to load
        LSLB                    get uP's version of ADDR
        LDY   DATAZONE          starting loc. of xpoint data
        ABY                     set up dynamic index offset
        LDX   $00,Y             get requested data
        STAA  PORTD             put command on portD
        STX   PORTC             dble. byte write ports C & B
* Disable latches and restore registers
        LDAA  #OFFPCDA          disable latch clocking
        STAA  PORTD             put command on portD
        STX   PORTC             dummy write to pulse STRB
        PULA                    restore registers
        PULB
        PULX
        PULY
        JMP   RETURN1           return to interrupt sequence
```

B.2.8 loaddata.doc

```
* LoadData sequence
* PRE: simple strobe mode; portD = output; all dev. disabled
* INLOAD = $00
* POST: same as PRE, but new DATA loaded into RAM

LOADDATA  PSHA                  save registers
          PSHB
          PSHX
          LDX   DATAZONE        index data into RAM
          STX   INFOZONE
          LDAA  #OUTMASK        configure portD as output port
          STAA  DDRD            put command on portD
          STAA  INLOAD          set INLOAD to all ones
          LDAB  #$D6            change to full input handshake
```

```
                STAB    PIOC        write to I/O control register
                CLI                 enables further int. from STRA
                LDAB    NEWADDR     load # of xpoint datum to get
                ASLB                mult. by 2 to get uP's version
                STAB    COUNT       store uP's ADDR values
STARTIMP        LDAA    #4          initialize IMPCOUNT to 4
                STAA    IMPCOUNT
DECIMP          SUBA    #1          decrement IMPCOUNT by 1
                STAA    IMPCOUNT    store new IMPCOUNT back to mem
                STAA    PORTD       put command onto portD
                SUBB    #2          decrement COUNT by 2
                STAB    COUNT       store new COUNT back to mem
                LDAB    PIOC        setup read
                LDAB    PORTCL      dummy read to trigger STRB

* Interrupt from STRA should be triggered and now RTIed
* Now check if finished
                LDAA    IMPCOUNT
                BNE     DECIMP      if IMPCOUNT<> 0 then decimp
                LDAB    COUNT
                BNE     STARTIMP    if COUNT <> 0 then startimp
* all data is now loaded into RAM
                LDAB    #$06        change I/O to simple strobe
                STAB    PIOC
                SEI                 disallow further IRQ ints
                PULA
                PULB
                PULX
                JMP     RETURN1     go back to INTERRUPT sequence
```

B.2.9 error.doc

```
* Error sequence
* PRE: Simple strobe, active low pulse; portD = output
* all dev. Hi-Z
* POST: same as pre, but error code returned in 4 MS bits
* of MS byte of DATA sent to PC
* note: portC is more sig than portB
ERROR   PSHA    save registers
        PSHB
        PSHX
        LDX     INFOZONE        prep. for indexing DATA in ram
        LDD     NEWDALO         dble. byte read in acc A & B
        ORAA    #$A0            error on top of MS data byte
        XGDX                    exchange D with X register
        LDAB    #ONPCDA         enable INTFC latches' clocking
        STAB    PORTD           put command on portD
        STX     PORTC           dble. byte write to port C & B
        LDAB    #OFFPCDA        disable INTFC latches
        STX     PORTC           dummy write to pulse STRB
        PULA                    restore registers
        PULB
        PULX
        JMP     RETURN1         return to interrupt sequence
```

Appendix C

Schematic Diagrams

This appendix contains the schematic diagrams for the IBM-PC to 68HC11 interface board, the output interface board, the D/A converter board, and the address decoding and sample-and-hold circuit board.

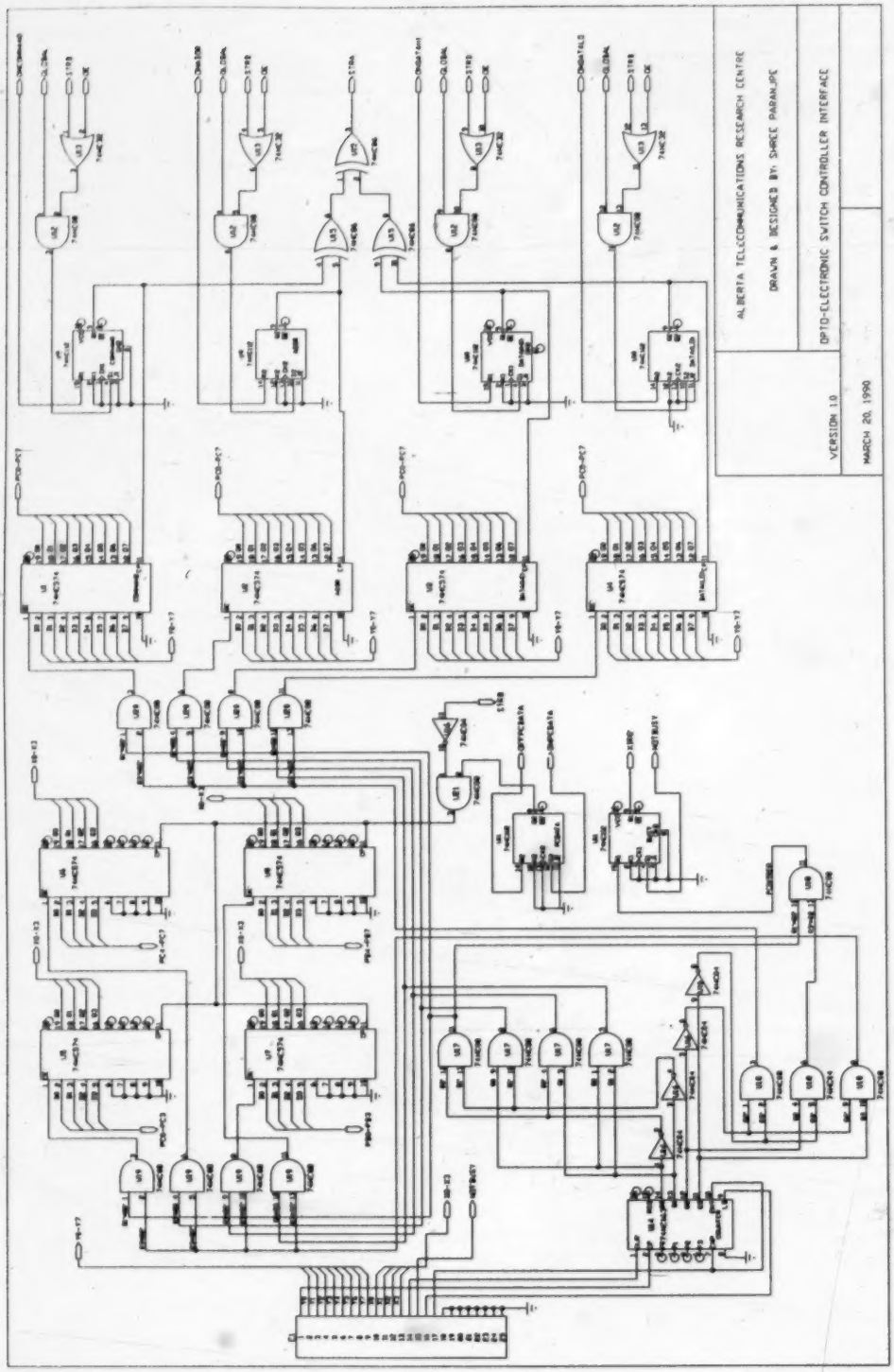

Figure C.1. *IBM-PC to 68HC11 interface board*

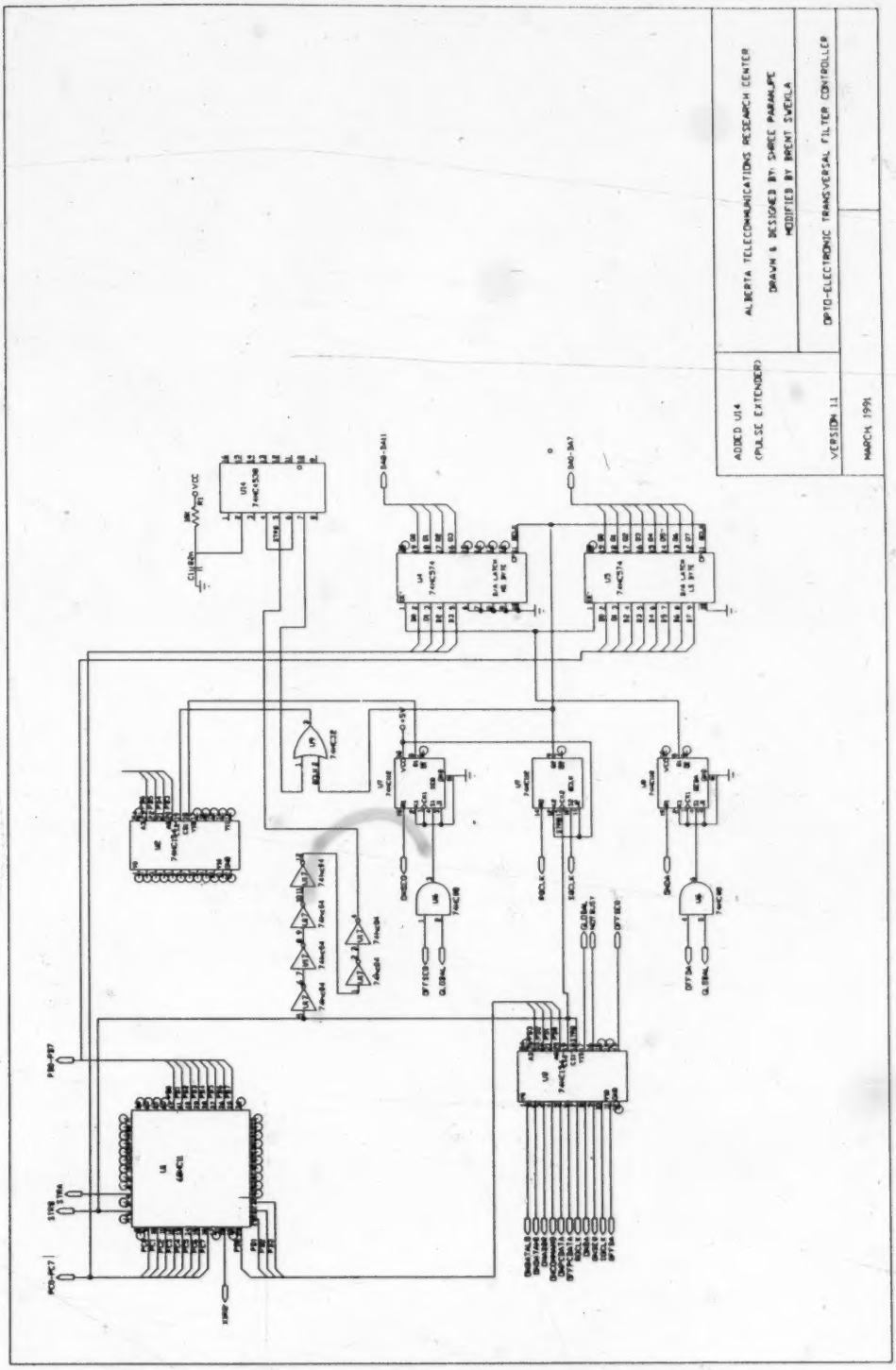

Figure C.2. *Output interface board*

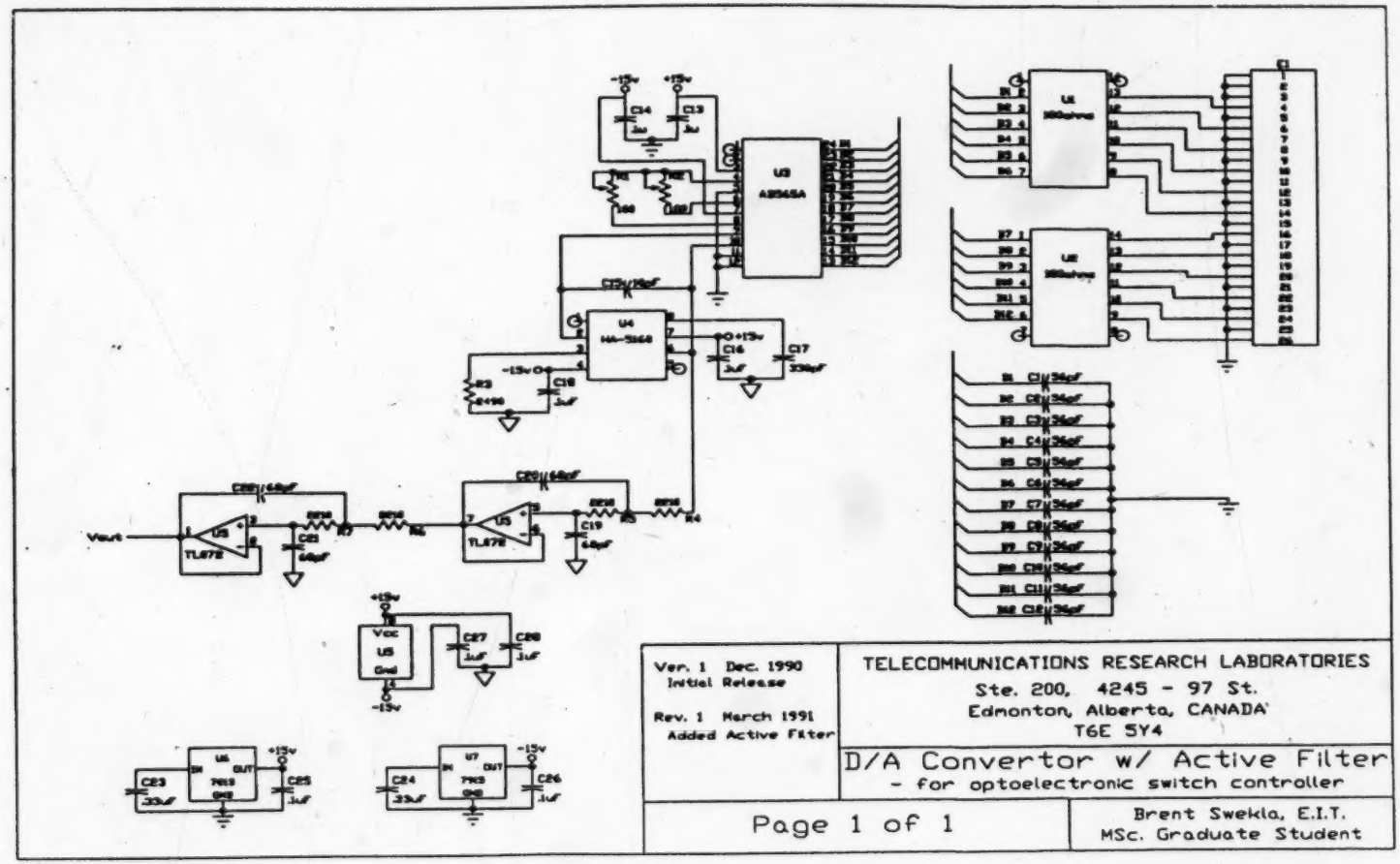

Figure C.3. *D/A converter board*

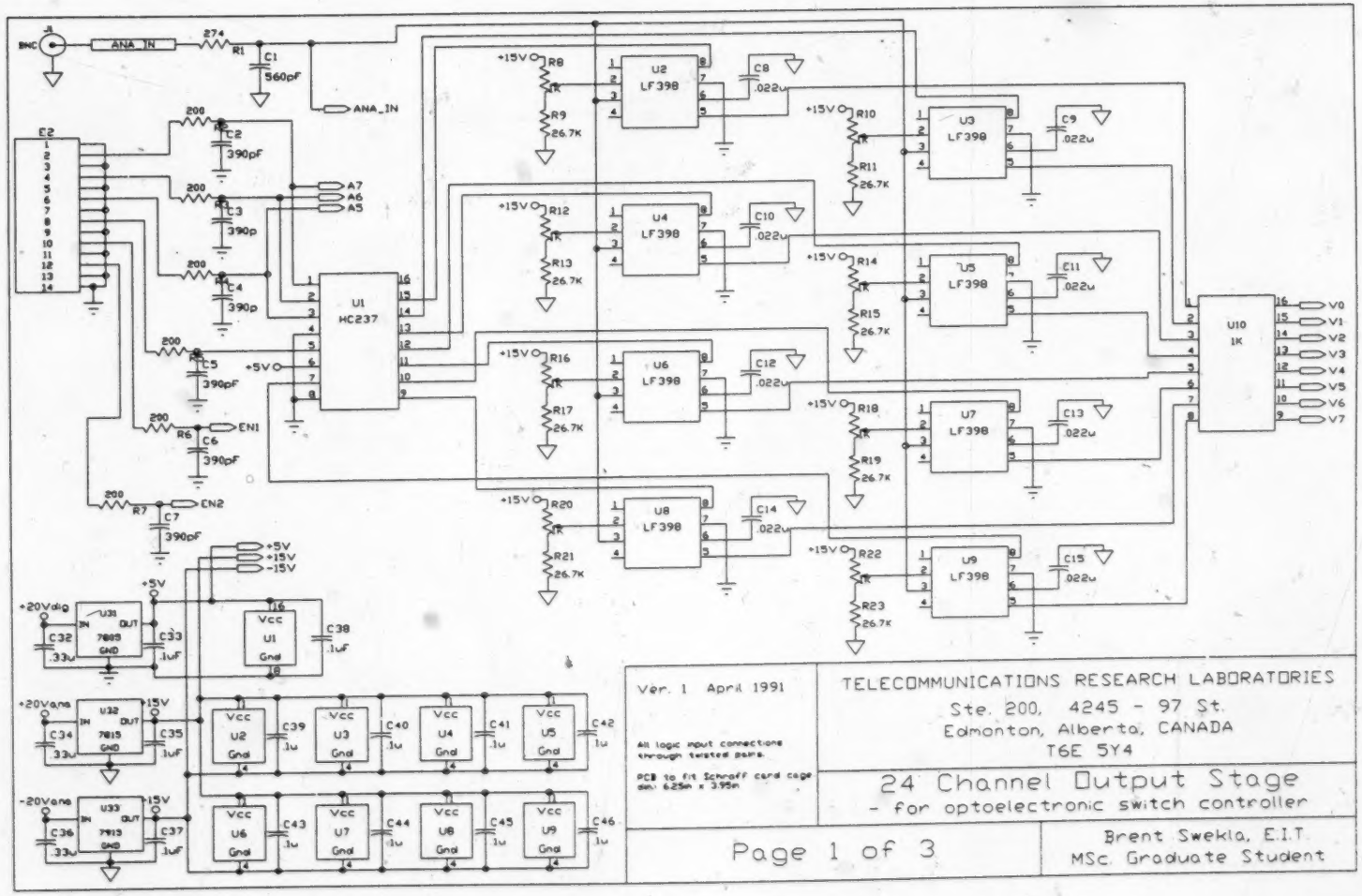

Figure C.4 (a) *Address decoding and sample-and-hold circuit board*

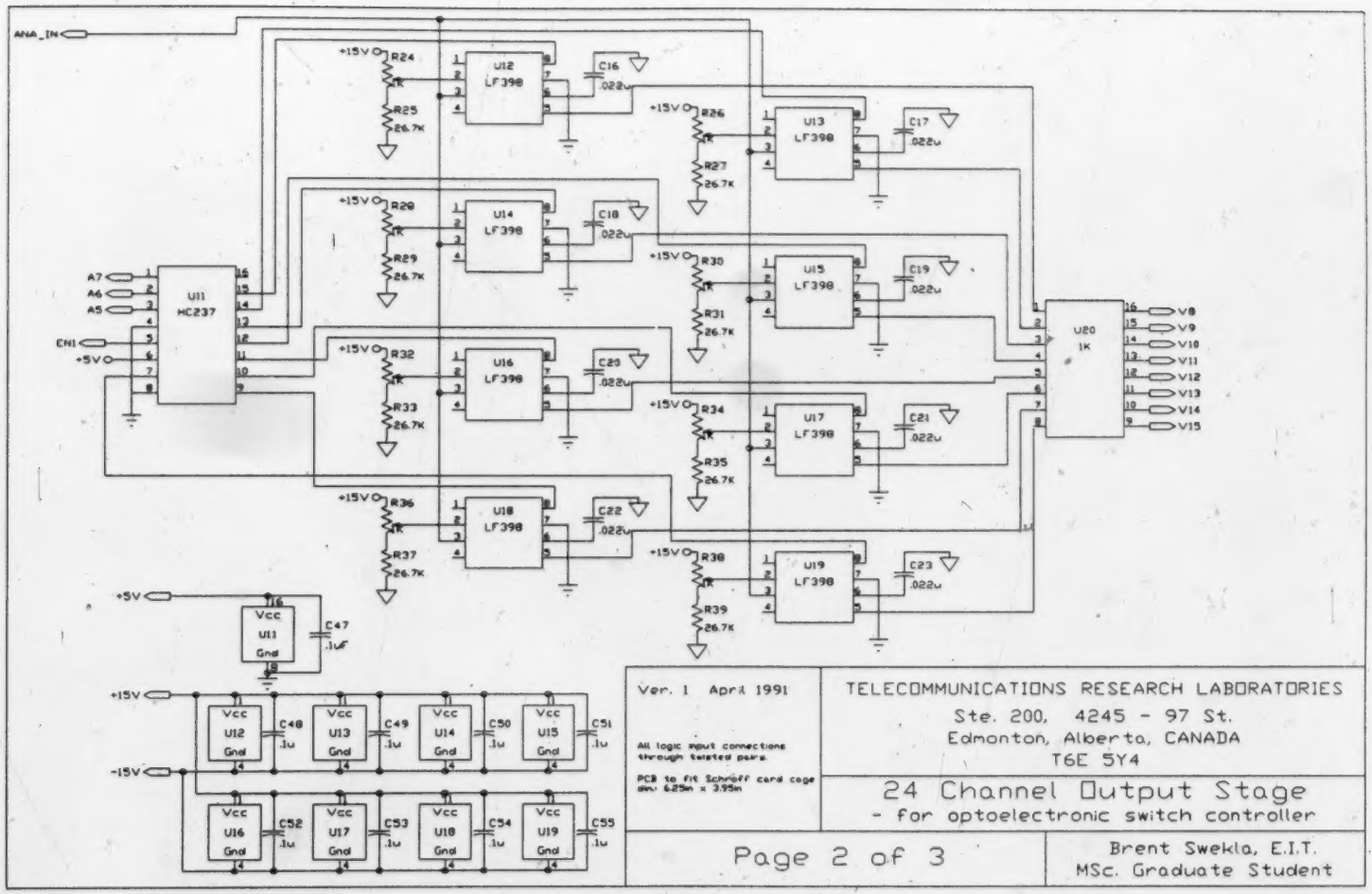

Figure C.4 (b) *Address decoding and sample-and-hold circuit board*

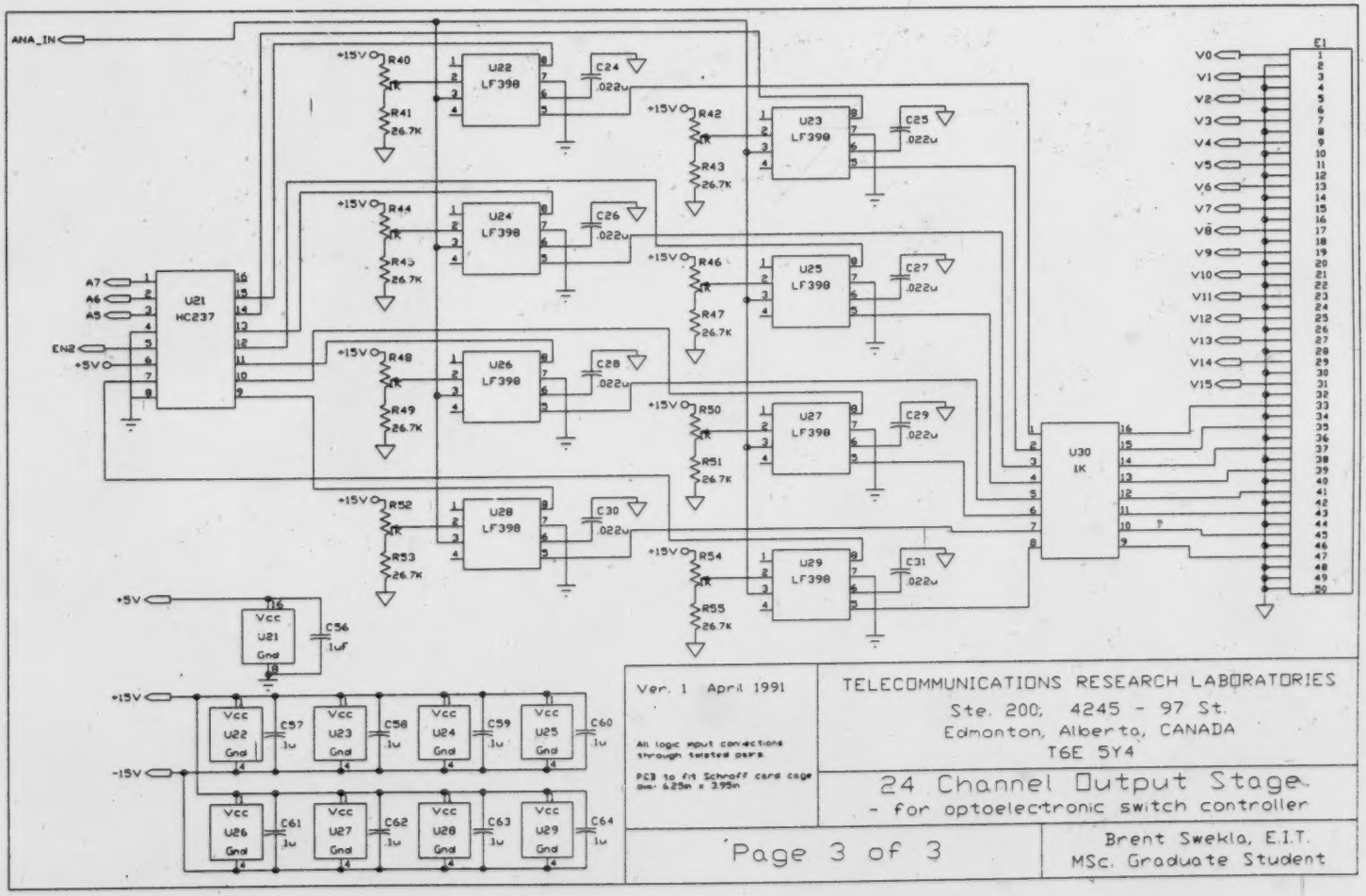

Figure C.4 (c) *Address decoding and sample-and-hold circuit board*

Appendix D

Multi-Channel Analog Voltage Controller Operating Instructions

This appendix contains instructions for compiling the controller software, downloading the assembled 68HC11 microcontroller software to the 68HC11 evaluation board, and executing the programs.

Compiling the Software

The 68HC11 microcontroller assembly language is compiled with a cross-assembler provided by Motorola called *as11.exe*. It is executed from DOS by typing "as11" followed by the names of the assembly language files in the order in which they appear in Table 3.3. This will produce an assembled file called *initial.s19* in s-record format.

The C language user interface program files may be compiled by any C compiler for the IBM-PC; in the present work Turbo-C was used to produce the executable file *ibm_oetf.exe*.

Downloading the Microcontroller Software

In the follwing instructions "<CR>" means press the return key and "<ctrl>" means press and hold the control key while typing the next character.

1. Run kermit.exe (type "kermit<CR>").
2. Type "set baud 9600<CR>" unless the baud rate is already set.
3. Ensure that the COM1 port of the PC is connected to the terminal port of the 68HC11 evaluation board.
4. Type "connect<CR>".
5. Apply power to the evaluation board or, if this has already been done, press the reset button on the evaluation board.
6. Type "<CR>". If the BUFFALO prompt appears, type "load t<CR>". If it doesn't, throw the board against the nearest wall; believe me, it deserves it.
7. Type "<ctrl>]c"; this should return you to kermit.
8. Type "type intial.s19>com1<CR>".

Executing the Software

1. From kemit type "connect<CR>"; this should return you to the microcontroller monitor.
2. Type "g C000<CR>" to begin execution of the microcontroller software (the first instruction is at hexadecimal address C000).
3. Type "<ctrl>]c" to return to kermit.
4. Type "quit<CR>" to return to DOS.
5. Type "ibm_oetf<CR>" to begin execution of the user interface.

Alternatively,

1. Acquire a Browning 28" .12 gauge pump action shot gun and 3" magnum shells with No. 4 steel shot.
2. To be sporting have someone throw the disk into the air.
3. Let'er rip!

Appendix E

Optoelectronic Transversal Filter

B.E.Swekla and R.I.MacDonald
TR*Labs*
4245 97th St, Edmonton,Alberta
Canada T6E 5Y7

Indexing Terms

Delay lines, Optical fibres, Signal processing

Abstract

We report the development of a 1 GHz bandwidth, 15 tap transversal filter with optical fibre delays and controllable bipolar weights set with variable sensitivity GaAs metal-semiconductor-metal photodetectors. The filter's performance is demonstrated by configuring it as tunable bandpass filter. The use of optoelectronically set bipolar weights represents a significant advance over previously demonstrated optical fibre transversal filters.

1. Introduction

The wideband, low loss signal propagation which is characteristic of optical fibres opens the possibility of using fibre delays to achieve very high speed signal processors / 1,2,3/. Intensity modulated signals on optical carriers can be delayed and combined to produce filters for signals with bandwidths characteristic of optical communications technology, that is, several GHz. Optical fibre transversal filters have been demonstrated in sequential configurations, using a single tapped fibre /2/, and parallel configurations, using a fibre power splitter to feed an array of delay lines /1/.

In these systems coherent summation of intensity modulated signals occurs with incoherent summation of the optical carrier power. The usual method of summing the delayed signals is simply to deliver the light from all delay lines to the same detector. This approach may be problematic. The efficient collection of light from multiple sources into a single detector is difficult for high bandwidths because of size constraints in the detector to maintain low capacitance. Tap weights must be set by varying the intensity of the light emanating from each delay line. This requires the system to include a device with controllable transmittance such as a spatial light modulator, which may impose significant excess loss, and

may also set constraints on the dynamic range of the weights, their resolution, and the speed with which they can be updated. Optical interference can occur in the carrier when single mode delay lines are used, if the delays are shorter than the source coherence length. While the delay- bandwidth requirements of many applications would permit multimode fibres, single mode propagation may be dictated by the method of controlling tap weights, as when optical directional couplers are used for this purpose /4/.

We report the demonstration of a wideband optoelectronic transversal filter with variable tap weights which are set by controlling the sensitivity of photodetectors. The device offers essentially the same advantages as other optical tapped delay line filters, and in addition provides controllable, bipolar tap weights, and is inexpensive to fabricate. The technique is an adaptation of the optoelectronic switching principle /5/, and demonstrates much of the technology needed for the synthesis of wideband optoelectronic neural networks /6/ .

2. Experiment

The configuration of our 16-tap transversal filter is shown in Fig. 1. The broadband electrical input signal modulates the drive current in a short wavelength (783 nm) pigtailed laser diode to produce an intensity modulated optical signal. This signal is split using a commercially available 1x16 optical fibre power splitter; optical fibre (50/125 µm graded index) delays in increments of 20 cm are fusion spliced to the output of the splitter. The light from each fibre is coupled to a single detector in a 1x16 detector array. The detectors used in this demonstration are interdigital metal-semiconductor-metal (MSM) GaAs photodiodes with active areas 100 µm square, and a channel spacing of 5 µm; they are fabricated in a linear array with a center-to-center spacing of 250 µm. The fibres are aligned with the detectors by first positioning them in v-grooves etched in a silicon wafer; the fibres are then epoxied in place and the entire v-groove wafer aligned with the detector array. With this technique uniformities of ±1.5 dB have been achieved in the detected electrical signal. The electrical signals from the different delay paths are then summed passively (one contact of each photodiode is common) and amplified to form the output.

By adjusting the dc bias voltage across an MSM photodiode we control its sensitivity. Typical sensitivity to a 100 MHz sine wave as a function of bias voltage is shown in Fig. 2 for the photodiode used. An important feature of signal summation after detection in optoelectronic filters of this type is the ability to implement bipolar weights. This is achieved with MSM detectors by reversing the polarity of the bias which introduces a 180° phase shift in the output /6,7/ .

It is desirable that the shape of the frequency response of the detector be independent of bias voltage (ie. all weights exhibit the same frequency response). The frequency response of our photodiodes at four different bias voltages is shown in Fig. 3; it is clear that the responsivity of the detectors drops off more steeply for the smaller bias voltages. Although this will degrade the performance of the filter, the effect is relatively minor over most of the frequency span.

The precision with which the weights can be set is also an important characteristic of optoelectronic delay-line filters /8/. One limitation is the resolution of the applied bias voltage. Typically there is a trade-off between reconfiguration speed and resolution; our multichannel analog voltage controller can reconfigure 15 taps in 750 μs with a worst case accuracy of 10 mV. Another limitation is the signal-to-noise ratio of the output signal. The MSM detector array is from an experimental process, and has a sensitivity of about 0.1 A/W. This relatively weak sensitivity limits us to a SNR of ~30 dB in a 1 GHz bandwidth; this restriction is of the same order as that produced by the control voltage resolution. A more serious precision limitation in our system is modal noise in the multimode fibre delays; this results in an uncertainty of 3-5%in the weight values.

3. Frequency Domain Fourier Synthesis

To demonstrate our system we have configured it as a variable bandpass filter. The impulse response of an ideal N-tap transversal filter is given by

$$h(t) = \sum_{n=1}^{N} w_n \delta(t - T_n) \tag{8}$$

where w_n are the weights, T_n are the delays and δ is the Dirac function. The corresponding transfer function is given by

$$H(f) = \sum_{n=1}^{N} w_n e^{-j2\pi n T_o f} \tag{9}$$

where uniform delay increments of T_o are assumed. Substituting $m = n - (N+1)/2$ and $M = (N-1)/2$ gives

$$H(f) = e^{-j(N+1)(\pi T_o) f} \sum_{m=-M}^{M} w_m e^{-jm(2\pi T_o) f} \tag{10}$$

where the first term is a linear phase term and the summation is a truncated complex Fourier series. The system demonstrated here is restricted to real weights. It is, however, possible to implement complex weights by using two systems in quadrature. Limiting our synthesis to linear phase filters and even frequency response functions implies that the weights will be real and $w_m = w_{-m}$. We have used a Gaussian window to minimize the effects of the Fourier series truncation.

Fig. 4 shows the optoelectronic synthesis of bandpass filters at two arbitrary center frequencies: 200 MHz and 350 MHz. The data are not corrected for the frequency responses of the laser, detectors, and amplifier. The passbands are clearly visible at the specified frequencies with ~15 dB sidelobe rejection,which may be compared with theoretical calculations indicating about 20 dB rejection. We believe that the principle source of performance degradation is nonuniformity in the delay increments as simulation studies have shown that the sidelobe structure is primarily sensitive to such nonuniformity. Other possible sources of degradation are the imprecision of the weights and optical crosstalk at the detector array. We anticipate that improved precision will be obtained in both delay increments and tap weights when the device is fabricated using hybrid optoelectronic integration techniques.

4. Conclusions

We have made the first successful demonstration of a transveral filter employing optical fibre delays and optoelectronically set bipolar weights. The active devices employed have the potential for operation up to about 10 GHz; with improvement in the precision of the passive delay lines, filters that are rapidly tunable over a very wide range are foreseeable. This work has applications in agile and adaptive wideband filters, equalizers, and neural networks.

5. Acknowledgements

We would like to acknowledge the contributions of the Communications Research Centre in fabricating the detectors, and the Alberta Microelectronic Centre in fabricating the v-grooves.

References

5 CHANG, C. T., CASSABOOM, J. A., and TAYLOR, H. F. : "Fibre-optic delay-line devices for r. f. signal processing", *Electron. Lett.*, 1977, **13**, (22), pp. 678-680

6 JACKSON, K. P., NEWTON, S. A., MOSLEHI, B., TUR, M., CUTLER, C. C., GOODMAN, J. W., and SHAW, H. J. : "Optical fiber delay-line signal processing", *IEEE Trans. Microwave Theory Tech.*, 1985, **MTT-33**, (3), pp. 193-210

7 MACDONALD, R. I. : "Switched optical delay-line signal processors", *J. Lightwave Technol.*, 1987, **LT-5**, (6), pp. 856-861

8 GOOKIN, D. M., and BERRY, M. H. : "Finite impulse response filter with large dynamic range and high sampling rate", *Appl. Opt.*, 1990, **29**, (8), pp. 1061-1062

9 MACDONALD, R. I. : "Optoelectronic matrix switching", *Canadian Journal of Physics*, 1988, **67**, pp. 389-393

10 MACDONALD, R. I., and LEE, S.S. : " Photodetector sensitivity control for weight-setting in optoelectronic neural networks", *Appl. Opt.*, 1991, **30**, pp. 176-178

11 OHTA, J., NITTA, Y., and KYUMA, K. : "Dynamic optical neurochip using variable-sensitivity photodiodes", *Opt. Lett.*, 1991, **16**, pp. 744-746

12 PAPARAO, P., GHOSH, A., and ALLEN, S. D. : "Design and performance optimization of fiber optic adaptive filters", *Appl. Opt.*, 1991, **30**, pp. 1826-1838

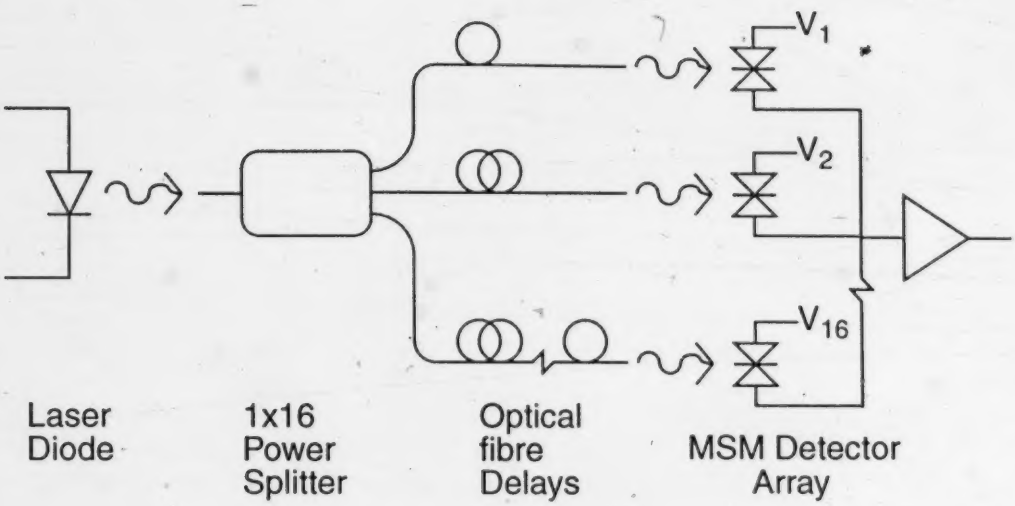

Figure E.1. *Optoelectronic transversal filter*

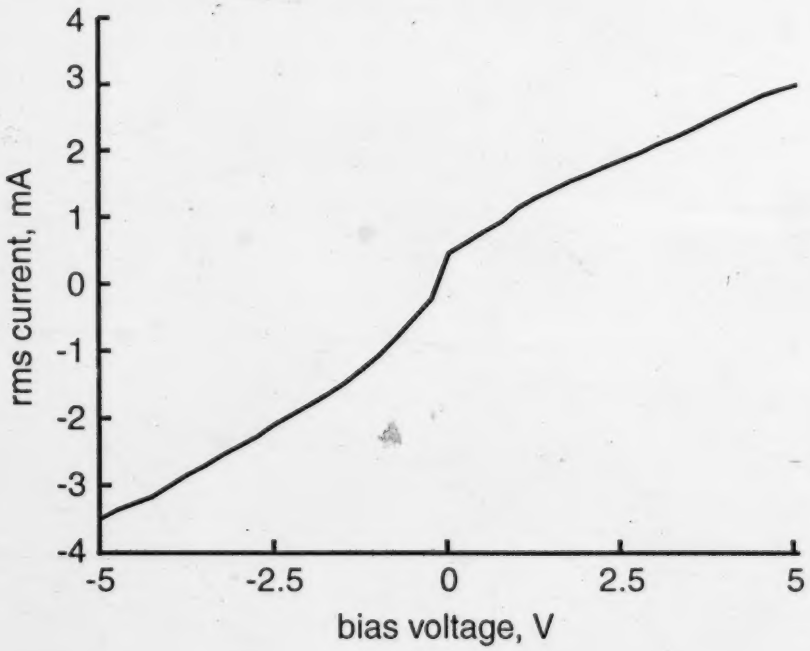

Figure E.2. *MSM sensitivity as a function of dc bias voltage*

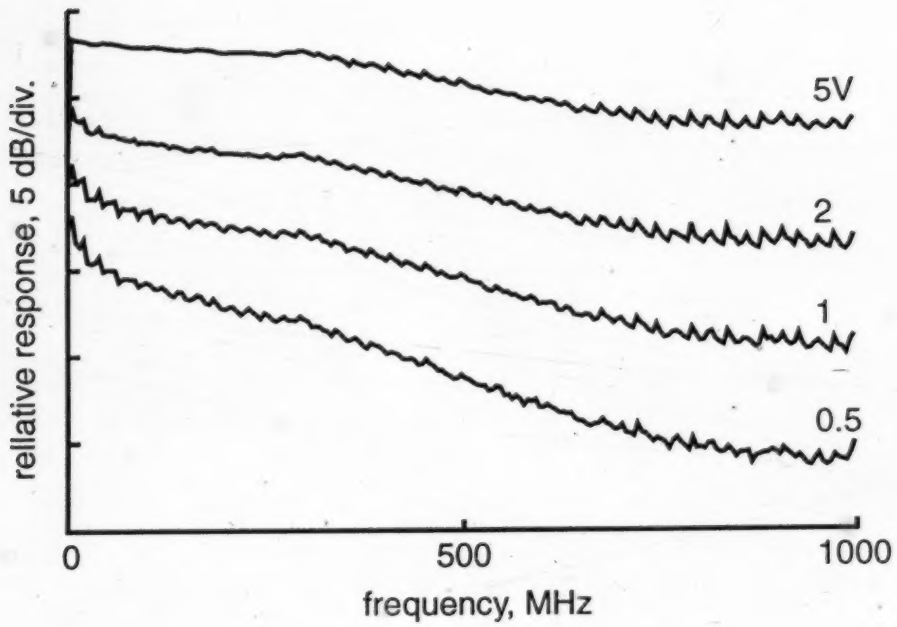

Figure E.3. *MSM frequency response for various bias voltages*

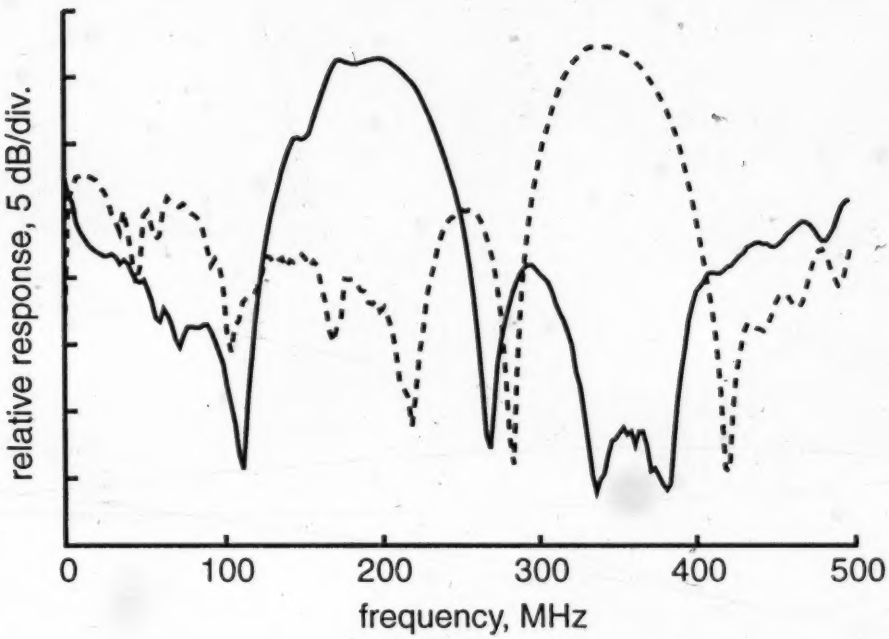

Figure E.4. *Bandpass filter synthesis with center frequencies of 200 MHz (solid line) and 350 MHz (dashed line)*